

3/5/1

DIALOG(R) File 351:Derwent WPI

(c) 2003 Thomson Derwent. All rts. reserv.

013681526 \*\*Image available\*\*

WPI Acc No: 2001-165739/ 200117

Method and apparatus for program update - NoAbstract

Patent Assignee: LG ELECTRONICS INC (GLDS )

Inventor: NAM C H

Number of Countries: 001 Number of Patents: 002

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
KR 2000040430	A	20000705	KR 9856063	A	19981218	200117 B
KR 284430	B	20010402	KR 9856063	A	19981218	200216

Priority Applications (No Type Date): KR 9856063 A 19981218

Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
-----------	------	-----	----	----------	--------------

KR 2000040430	A			H04L-012/16	
---------------	---	--	--	-------------	--

KR 284430	B			H04L-012/16	Previous Publ. patent KR 2000040430
-----------	---	--	--	-------------	-------------------------------------

Title Terms: METHOD; APPARATUS; PROGRAM; UPDATE; NOABSTRACT

Derwent Class: W01

International Patent Class (Main): H04L-012/16

File Segment: EPI

3/39/1

DIALOG(R) File 345:Inpadoc/Fam.& Legal Stat  
(c) 2003 EPO. All rts. reserv.

18295426

Basic Patent (No,Kind,Date): KR 2000040430 A 20000705 <No. of Patents:  
.001>

Patent Family:

Patent No	Kind	Date	Applic No	Kind	Date
KR 2000040430	A	20000705	KR 9856063	A	19981218 (BASIC)

Priority Data (No,Kind,Date):  
KR 9856063 A 19981218

PATENT FAMILY:

KOREA, REPUBLIC (KR)

Patent (No,Kind,Date): KR 2000040430 A 20000705  
APPARATUS AND METHOD FOR UPDATING PROGRAM (English)  
Patent Assignee: LG ELECTRONICS INC  
Author (Inventor): NAM CHA HEE (KR)  
Priority (No,Kind,Date): KR 9856063 A 19981218  
Applic (No,Kind,Date): KR 9856063 A 19981218  
IPC: \* H04L-012/16  
Language of Document: Korean

공개특허특 1999-030334

Corresponding to  
JP11-110141

(19) 대한민국특허청(KR)  
(12) 공개특허공보(A)

(51) Int. Cl.<sup>6</sup> (11) 공개번호 특1999-030334  
G06F 12/00 (43) 공개일자 1999년04월26일

(21) 출원번호 특1998-041922  
(22) 출원일자 1998년09월30일  
(30) 우선권주장 97-267177 1997년09월30일 일본(JP)  
(71) 출원인 소니 가부시끼 기이샤 이데이 노부유키  
일본국 141-0001, 도쿄도, 시나가와쿠, 기타시나가와 6초메, 7반, 35고  
(72) 발명자 후세 히로아키  
일본 도쿄도 시나가와쿠 기타시나가와 6-7-35 소니(주)내  
사사 아키라  
일본 도쿄도 시나가와쿠 기타시나가와 6-7-35 소니(주)내  
(74) 대리인 이병호

심사청구 : 없음

(54) 외부 기억 장치 및 데이터 처리 방법

요약

본 발명에 따른 외부 기억 장치는 소정의 블록 단위로 데이터를 소거하도록 배열된 플래시 메모리를 사용하고 부트 데이터가 판독될 때 존재하는 에러가 방지되도록 하는 방식으로 구성된다. 더욱이, 부트 데이터가 기억된 각각의 블록내에 기억된 부트 데이터가 새로운 것인지 오래된 것인지를 나타내는 식별 번호가 각각의 블록내에 기억된다. 외부 기억 장치가 부트 업될 때, 다수의 다른 블록내에 기억된 부트 데이터중에서 최신 부트가 식별 번호를 따라 판독된다. 최신 부트 데이터는 외부 기억 장치를 부트 업하기 위해 사용된다. 외부 기억 장치가 부트 업될 때, 다수의 다른 블록내에 기억된 부트 데이터가 새로운 것인지 오래된 것인지를 식별 번호에 따라 판정한다. 만일, 오래된 부트 데이터가 존재한다면, 오래된 부트 데이터는 최신 부트 데이터로 재기록된다.

도면

도8

명세서

도면의 간단한 설명

- 도 1은 본 발명이 적용된 시스템의 전체 구조를 도시하는 도면.
- 도 2는 본 발명이 적용된 메모리 카드의 구조를 도시하는 블록도.
- 도 3은 본 발명이 적용된 메모리 카드의 형태를 도시하는 사시도.
- 도 4는 본 발명이 적용된 메모리 카드의 기억 영역의 구조를 도시하는 도면.
- 도 5는 분산 관리 정보의 구조를 도시하는 도면.
- 도 6은 추가의 관리 정보의 구조를 도시하는 도면.
- 도 7은 각각의 블록의 분산 관리 정보로부터 구성된 집합 관리 정보에서의 상태를 도시한 도면.
- 도 8은 부트 블록을 갱신하는 과정을 도시하는 도면.
- 도 9는 메모리 카드가 부트 업될 때 부트 블록을 판독하는 과정을 도시하는 흐름도.
- 도 10은 메모리 카드가 부트 업될 때 부트 블록을 판독하는 과정을 도시하는 흐름도.
- 도 11은 메모리 카드가 부트 업될 때 부트 블록을 판독하는 과정을 도시하는 흐름도.
- 도 12는 메모리 카드가 부트 업될 때 부트 블록을 판독하는 과정을 도시하는 흐름도.

\*도면의 주요 부분에 대한 상세한 설명\*

- 1 : 데이터 처리 장치
- 2 : 메모리 카드
- 4 : 내부 메모리
- 5 : 보조 기억 장치

본 발명의 상세한 설명

## 발명의 목적

### 발명이 속하는 기술 및 그 분야의 종래기술

#### 발명의 분야

본 발명은 소정의 블록 단위로 데이터를 소거하도록 배열된 기억 수단을 사용하는 외부 기억 장치 및, 이러한 형태의 외부 기억 장치에 기억된 데이터를 처리하는 방법에 관한 것이다.

#### 관련된 배경 기술

개인용 컴퓨터 또는 디지털 스틸 카메라 등의 데이터 처리 장치에 사용하는 형태의 외부 기억 장치, 이른바 플래시 메모리를 사용하는 형태의 외부 기억 장치가 공지된다.

플래시 메모리를 사용하는 외부 기억 장치는 데이터 영역을 블록 단위로 관리하기 위해 다수의 블록으로 플래시 메모리의 기억 영역이 분할되는 구조를 갖는다. 즉, 블록 단위로 데이터의 소거가 실행된다. 만일, 복구 불가능한 에러가 기억 영역내에 생기면, 에러가 있는 영역을 포함하는 블록의 사용은 금지된다. 상기 형태의 블록은 사용 금지 블록으로 불린다.

상기 형태의 외부 기억 장치는 외부 기억 장치에 대한 액세스가 요구된 정보를 포함하는 부트 데이터가 소정의 블록내에 미리 기억되는 구조를 갖는다. 외부 기억 장치가 부트 업될 때, 부트 데이터는 초기에 판독된다. 그후, 액세스는 데이터가 기록/판독되도록 얻어진 부트 데이터에 따라 외부 기억 장치에 대해 이루어진다.

상기 형태의 외부 기억 장치에서, 부트 데이터는 외부 기억 장치에 대한 액세스에 요구된 정보를 일반적으로 포함한다. 만일, 부트 데이터가 판독될 수 없다면, 외부 기억 장치에 대한 액세스는 이루어지지 않는다. 따라서, 부트 데이터는 만족스런 신뢰도를 가져야 한다.

### 발명이 이루고자 하는 기술적 과제

종래, 부트 데이터가 기억된 블록(이하 부트 블록)은 소정의 위치에서 형성된 블록이다. 일반적으로 사용된 부트 블록이 사용 금지 블록으로 사용될 지라도, 부트 데이터의 카피는 부트 데이터가 판독되도록 또다른 블록내에 저장된다. 종래의 외부 기억 장치에서, 신뢰도는 부트 데이터의 다수의 카피를 준비함으로써 또한 개선된다. 더욱이, 카피는 다른 블록내에 기억된다. 그러나, 부트 데이터의 다수의 카피가 준비되는 방법은 기억 용량을 효율적으로 사용하는 면에서 바람직하지 않은 방법이다.

종래, 부트 데이터가 갱신될 때, 부트 데이터의 모든 카피가 최신 부트 데이터로 갱신될지 아닐지가 판정된다. 따라서, 만일, 부트 데이터의 카피가 소정의 동작에서 최신 부트 데이터로 정확하게 갱신되지 않는다면, 즉, 최신의 정확한 부트 데이터는 판독될 수 없다. 상기 경우, 외부 기억 장치에 대한 정확한 액세스가 이루어질 수 없다.

종래, 부트 데이터의 카피가 준비될지라도, 그 카피는 원래의 부트 데이터의 판독이 실패할 때만 사용된다. 따라서, 원래의 부트 데이터가 저장되어 바람직하지 않게 반전되는 부트 블록 부분의 비트를 때문에, 원래의 부트 데이터의 내용이 부정확하게 이루어진다면, 상기 부트 데이터가 판독될 수 있을 때 상기 부트 데이터가 사용되는 문제가 발생한다.

### 발명의 구성 및 작용

#### 발명의 개요

본 발명의 목적은 기억 영역을 효율적으로 사용할 수 있고, 부트 데이터가 판독될 때 에러가 쉽게 생기지 않으며, 만족스런 신뢰도를 갖는 외부 기억 장치를 제공하는 것이다.

본 발명의 또다른 목적은 상기 외부 기억 장치가 실행될 수 있는 데이터 처리 방법을 제공하는 것이다.

본 발명에 따른 한 양상에 따라, 데이터의 소거를 소정의 블록 단위로 행함과 함께, 부트 업시에 최초로 판독된 부트 데이터를 기억하는 블록을 갖는 외부 기억 장치가 제공된다. 부트 데이터는 다수의 다른 블록에 각각 기억된다. 더욱이, 부트 데이터가 기억된 각 블록에 기억된 부트 데이터가 새로운 것인지 오래된 것인지 나타내는 식별 번호가 각각의 블록에 기억된다.

프로그램에 따른 외부 기억 장치는, 다수의 다른 블록내에 기억된 부트 데이터중에서 최신의 부트 데이터가 식별 번호에 따라 판독된다. 그후, 최신 부트 데이터는 상기 외부 기억 장치가 부트 업되도록 사용된다.

본 발명에 따른 외부 기억 장치는, 외부 기억 장치가 부트 업될 때, 식별 번호에 따라, 다수의 다른 블록내에 기억된 각각의 부트 데이터가 새로운 것인지 오래된 것인지를 판정한다. 오래된 부트 데이터가 존재하는 경우, 오래된 부트 데이터는 최신 부트 데이터로 재기록된다.

본 발명의 또다른 양상에 따라, 소정의 블록 단위로 데이터를 소거하도록 배열된 외부 기억 장치가 부트 업될 때 최초 판독된 부트 데이터는 상기 외부 기억 장치내에 기억되는 데이터 처리 방법이 제공된다. 이 방법은 다수의 다른 블록의 각각의 블록에 부트 데이터를 기억하는 단계; 및 부트 데이터가 기억된 각각의 블록내에, 그 블록내에 기억된 부트 데이터가 새로운 것인지 오래된 것인지를 나타내는 식별 번호를 기억하는 단계를 구비한다.

본 발명에 따른 데이터 처리 방법은, 상기 외부 기억 장치가 부트 업될 때, 다수의 다른 블록내에 기억된 부트 데이터중에서 최신 부트 데이터가 식별 번호에 따라 판독되고, 외부 기억 장치는 최신 부트 데이터를 사용하여 부트 업된다.

본 발명에 따른 데이터 처리 방법은, 상기 외부 기억 장치가 부트 업될 때, 다수의 다른 블록내에 기억된 부트 데이터가 새로운 것인지 오래된 것인지를 식별 번호에 따라 판정하여, 오래된 부트 데이터가 존재할 때 오래된 부트 데이터는 최신 부트 데이터로 재기록되는 구조를 갖는다.

본 발명의 다른 목적, 특징, 장점은 도면을 참조하여 양호한 실시예의 상세한 설명으로부터 명백해질 것이다.

#### 양호한 실시예의 설명

본 발명의 실시예는 도면을 참조하여 설명될 것이다.

도 1은 본 발명이 적용된 시스템의 일예의 전체 구성을 도시한 도면이다. 상기 실시예에 따른 시스템은 직렬 인터페이스를 통해 데이터 처리 장치(1)에 접속된 외부 기억 장치인 메모리 카드(2) 및 호스트 시스템이 있는 데이터 처리 장치(1)를 사용하고 있다.

데이터 처리 장치(1)와 메모리 카드(2)간에 직렬 인터페이스를 통해 데이터가 통신되는 구조를 갖는 시스템이 설명될지라도, 본 발명은 데이터 통신이 병렬 인터페이스를 통해 실행되는 유형의 시스템에 적용될 수도 있다.

데이터 처리 장치(1)는 중앙 처리 장치(CPU)(3), 내부 메모리(4), 보조 기억 장치(5), 및 직렬 인터페이스 회로(6)를 포함한다. 상기 소자들은 버스(7)를 통해 상호 접속된다. 데이터 처리 장치(1)는 예를 들어, 내부 메모리(4)를 작업 영역으로서 사용함으로써 CPU(3)가 프로그램을 실행하도록, 보조 기억 장치(5)내에 기억된 프로그램을 판독한다. 이때, 데이터는 필요에 따라, 직렬 인터페이스 회로(6)를 통해 메모리 카드(2)에 통신된다.

본 발명에 따른 시스템에서 사용하는 데이터 처리 장치는, 그 데이터 처리 장치가 외부 기억 장치에 데이터를 통신할 수 있다면, 특히 한정되지 않는다. 본 발명은, 개인용 컴퓨터, 디지털 스틸 카메라, 및 디지털 비디오 카메라 등의 여러 데이터 처리 장치중의 한 장치에 적용될 수도 있다.

데이터 처리 장치(1) 및 메모리 카드(2)는 직렬 인터페이스를 통해 상호 접속된다. 구체적으로, 접속은 3개의 데이터 라인 SCLK, State, DIO를 통해 설정된다. 즉, 데이터 처리 장치(1) 및 메모리 카드(2)는 적어도, 데이터가 전송될 때 클록 신호를 전송하는 제 1 데이터 라인 SCLK, 데이터 전송이 실행될 때 요구된 상태 신호를 전송하는 데이터 라인 State, 및 메모리 카드(2)에 기록 및 판독되어야 하는 데이터를 직렬 전송하는 제 3 데이터 라인 DIO를 통해 상호 접속된다. 이와같이, 데이터는 데이터 처리 장치(1)와 메모리 카드(2)간에서 상기 데이터 라인을 통해 통신된다.

데이터는 일반적으로 데이터 처리 장치(1)와 메모리 카드(2)간에서, 헤더와 실제 데이터로 구성된 파일 단위로 통신된다. 예를 들어, 파일을 액세스하게 하는 정보, 데이터 처리 장치(1)에 의해 실행된 프로그램에 의해 요구된 정보 등은 파일의 헤더내에 기억된다.

도 2에 도시된 대로, 메모리 카드(2)는 소위 제어 IC로 구성되는 제어기(11) 및 이 제어기(11)에 의해 제어되는 플래시 메모리(12)를 포함한다.

제어기(11)는 직렬/병렬 변환 및 병렬/직렬 변환을 실행하는 직렬/병렬 및 병렬/직렬 인터페이스 시퀀서(13)(이하, S/P P/S 인터페이스 시퀀서(13)); 플래시 메모리(12)에 대한 액세스가 이루어질 수 있는 인터페이스로서 기능하는 플래시 메모리 인터페이스 시퀀서(14); S/P P/S 인터페이스 시퀀서(13)와 플래시 메모리 인터페이스 시퀀서(14)간에 통신되는 데이터를 일시적으로 기억하는 페이지 버퍼(15); 여러 정정 처리를 실행하는 여러 정정 회로(16); 플래시 메모리(12)에 대한 액세스를 제어하는 제어 명령의 발생을 실행하는 명령 발생기(17); 메모리 카드(2)의 버전 정보, 여러 속성 정보 항목 등이 기억된 컨피규레이션 ROM(18); 및 상기 각각의 회로의 동작에 요구된 클록 신호를 공급하는 발진기(19)를 포함한다.

S/P P/S 인터페이스 시퀀서(13)는 상기 3개의 데이터 라인 SCLK, State, DIO를 통해 데이터 처리 장치(1)의 직렬 인터페이스 회로(6)에 접속된다. 이와같이, S/P P/S 인터페이스 시퀀서(13)는 상기 데이터 라인 SCLK, State, DIO를 통해 데이터를 데이터 처리 장치(1)에 통신한다. 즉, S/P P/S 인터페이스 시퀀서(13)는 페이지 버퍼(15)로부터 전송된 데이터를 직렬 데이터로 변환하여 데이터 처리 장치(1)의 직렬 인터페이스 회로(6)에 상기 직렬 데이터를 전송한다. S/P P/S 인터페이스 시퀀서(13)는 데이터 처리 장치(1)의 직렬 인터페이스 회로(6)로부터 전송된 직렬 데이터를 병렬 데이터로 변환하여 상기 병렬 데이터를 페이지 버퍼(15)에 전송한다.

S/P P/S 인터페이스 시퀀서(13)와 데이터 처리 장치(1)간의 데이터 직렬 통신은, 동기가 제 1 데이터 라인 SCLK를 통해 데이터 처리 장치(1)로부터 공급된 클록 신호에 응답하여 유지되는 동안, 제 3 데이터 라인 DIO를 통해 실행된다. 제 3 데이터 라인 DIO를 통해 통신된 직렬 데이터의 유형은 제 2 데이터 라인 State를 통해 전송되는 상태 신호에 따라 결정된다. 직렬 데이터는 예를 들어, 플래시 메모리(12)상에 기록되는 데이터, 플래시 메모리(12)로부터 판독되는 데이터, 메모리 카드(2)등의 동작을 제어하는 제어 데이터로 분류된다. 상태 신호는 메모리 카드(2)의 상태를 나타내는데 사용된다. 상태 신호에 응답하여 나타낼 수 있는 메모리 카드(2)의 상태는, 데이터 처리 장치(1)로부터 메모리 카드(2)에 입력된 데이터가 소정의 처리동안 허가되지 않는 상태; 및 메모리 카드(2)내의 처리가 완료된 후 데이터 처리 장치(1)로부터 입력된 데이터가 대기되는 상태를 포함한다.

데이터 처리 장치(1)로부터 전송된 데이터가 메모리 카드(2)의 동작을 제어하는 제어 데이터일 때, S/P P/S 인터페이스 시퀀서(13)는 상기 제어 데이터를 명령 발생기(17)에 공급한다.

S/P P/S 인터페이스 시퀀서(13)를 통해 데이터 처리 장치(1)로부터 공급된 제어 데이터에 따라, 명령 발생기(17)는, 제어 명령을 플래시 메모리 인터페이스 시퀀서(14)에 전송하도록, 플래시 메모리(12)에 대한 액세스를 제어하는 제어 명령을 발생한다. 후술된 바와 같이, 플래시 메모리 인터페이스 시퀀서(14)는 제어 명령에 따라 플래시 메모리(12)상에 데이터를 기록하고 플래시 메모리(12)로부터 데

이터를 판독한다.

에러 소거 방지 스위치(20)는 명령 발생기(17)에 접속된다. 에러 소거 방지 스위치(20)가 스위치 온될 때, 명령 발생기(17)는, 플래시 메모리(12)상에 기록된 데이터의 소거를 명령하는 제어 데이터가 데이터 처리 장치(1)로부터 전송될지라도, 플래시 메모리(12)상에 기록된 데이터를 소거하는 제어 명령을 발생하지 않는다. 즉, 메모리 카드(2)는, 플래시 메모리(12)에 기억된 데이터가 에러 소거 방지 스위치(20)에 의해 소거될 수 없는 상태 또는 플래시 메모리(12)내에 기억된 데이터가 소거될 수 있는 상태중의 어느 한 상태로 될 수 있다.

S/P P/S 인터페이스 시퀀서(13)와 플래시 메모리 인터페이스 시퀀서(14)간에 위치한 페이지 버퍼(15)는, S/P P/S 인터페이스 시퀀서(13)와 플래시 메모리 인터페이스 시퀀서(14)간에 통신된 데이터를 일시적으로 기억하기 위해 버퍼 메모리이다.

S/P P/S 인터페이스 시퀀서(13)로부터 플래시 메모리 인터페이스 시퀀서(14)에 전송된 데이터는 S/P P/S 인터페이스 시퀀서(13)로부터 페이지 버퍼(15)에 초기 전송되며, 페이지 버퍼(15)내에 일시적으로 기억된다. 페이지 버퍼(15)내에 기억된 데이터에는 에러 정정 회로(16)에 의해 에러 정정 부호가 제공된다. 에러 정정 부호가 제공된 데이터는, 소정의 페이지 단위로(예를 들어, 1 페이지 = 512 바이트)로, 페이지 버퍼(15)로부터 플래시 메모리 인터페이스 시퀀서(14)에 전송된다.

이에 대한 대안으로서, 플래시 메모리 인터페이스 시퀀서(14)로부터 S/P P/S 인터페이스 시퀀서(13)에 전송된 데이터는 플래시 메모리 인터페이스 시퀀서(14)로부터 페이지 버퍼(15)에 초기 전송되며 페이지 버퍼(15)에 일시적으로 기억된다. 이때, 페이지 버퍼(15)에 기억된 데이터는 에러 정정 회로(16)에서 에러 정정 처리된다. 에러 정정 처리된 데이터는, 소정의 페이지 단위로, 페이지 버퍼(15)로부터 S/P P/S 인터페이스 시퀀서(13)에 전송된다.

명령 발생기(17)로부터 공급된 제어 명령에 따라, 플래시 메모리 인터페이스 시퀀서(14)는 플래시 메모리(12)상의 데이터 기록, 플래시 메모리(12)로부터의 데이터 판독등을 실행한다. 즉, 플래시 메모리 인터페이스 시퀀서(14)는 명령 발생기(17)로부터 공급된 제어 명령에 따라 플래시 메모리(12)로부터 데이터를 판독하여, 상기 설명된 대로, 페이지 버퍼(15)를 통해 상기 데이터를 S/P P/S 인터페이스 시퀀서(13)에 전송한다. 이에 대한 대안으로서, 명령 발생기(17)는 명령 발생기(17)로부터 공급된 제어 명령에 따라 페이지 버퍼(15)를 통해 S/P P/S 인터페이스 시퀀서(13)로부터 공급된 데이터를 수신하여, 플래시 메모리(12)상에 상기 데이터를 기록한다.

플래시 메모리(12)상에 기록되어야 하는 지령 데이터 형태로 데이터가, 데이터 처리 장치(1)로부터 3개의 데이터 라인 SCLK, State, DIO를 통해 메모리 카드(2)에 공급될 때, S/P P/S 인터페이스 시퀀서(13)는 먼저 지령 데이터를 병렬 데이터로 변환하여, 이 병렬 데이터를 상기 페이지 버퍼(15)에 전송한다. 페이지 버퍼(15)는 S/P P/S 인터페이스 시퀀서(13)로부터 전송된 데이터를 일시적으로 기억한다. 이때, 페이지 버퍼(15)에 기억된 데이터는 에러 정정 회로(16)에 의해 에러 정정 부호가 제공된다. 에러 정정 부호가 제공된 데이터는 소정의 페이지 단위로 플래시 메모리 인터페이스 시퀀서(14)에 전송된다. 명령 발생기(17)로부터 공급된 제어 명령에 따라, 플래시 메모리 인터페이스 시퀀서(14)는 페이지 버퍼(15)로부터 공급된 데이터를 플래시 메모리(12)상에 기록한다. 상기 언급된 처리의 결과, 데이터 처리 장치(1)로부터 전송된 데이터는 플래시 메모리(12)상에 기록된다.

메모리 카드(2)로부터 데이터가 판독될 때, 데이터는 먼저 명령 발생기(17)로부터 공급된 제어 명령에 따라 플래시 메모리 인터페이스 시퀀서(14)에 의해 플래시 메모리(12)로부터 판독된다. 플래시 메모리 인터페이스 시퀀서(14)는 플래시 메모리(12)로부터 판독된 데이터를 페이지 버퍼(15)에 전송한다. 페이지 버퍼(15)는 플래시 메모리 인터페이스 시퀀서(14)로부터 공급된 데이터를 일시적으로 기억한다. 이때, 페이지 버퍼(15)에 기억된 데이터는 에러 정정 회로(16)에서 에러 정정 처리된다. 에러 정정 처리된 데이터는, 소정의 페이지 단위로, S/P P/S 인터페이스 시퀀서(13)에 전송된다. S/P P/S 인터페이스 시퀀서(13)는 페이지 버퍼(15)로부터 공급된 데이터를 직렬 데이터로 변환하고, 그 후, 직렬 데이터를 3개의 데이터 라인 SCLK, State, DIO를 통해 데이터 처리 장치(1)에 전송한다. 상기 언급된 처리의 결과로서, 플래시 메모리(12)로부터 판독된 데이터는 데이터 처리 장치(1)에 전송된다.

데이터가 기록 또는 판독될 때, 데이터의 통신을 제어하는 제어 데이터는 데이터 처리 장치(1)로부터 메모리 카드(2)의 S/P P/S 인터페이스 시퀀서(13)에 전송되며 또한 플래시 메모리(12)상에 기록되어야 하는 데이터 또는 플래시 메모리(12)로부터 판독된 데이터는 통신된다. 상기 제어 데이터는 S/P P/S 인터페이스 시퀀서(13)로부터 명령 발생기(17)에 전송된다. S/P P/S 인터페이스 시퀀서(13)로부터 전송된 제어 데이터에 따라, 명령 발생기(17)는 플래시 메모리(12)에 대한 액세스를 제어하는 제어 명령을 발생한다. 제어 명령은 플래시 메모리 인터페이스 시퀀서(14)에 공급된다. 제어 명령에 따라, 플래시 메모리 인터페이스 시퀀서(14)는 데이터를 기록 또는 판독하기 위해 플래시 메모리(12)에 대해 액세스한다.

3개의 데이터 라인 SCLK, State, DIO에 추가하여, 메모리 카드(2)에는, 전원 라인 및 통상의 상태에서는 사용되지 않는 리저브 라인이 제공될 수도 있다. 예를 들어, 도 2 및 도 3은, 상술된 3개의 데이터 라인 SCLK, State, DIO에 추가하여, 4개의 전원 라인 VSS1, VSS2, VCC, INT, 및 3개의 리저브 라인 RSV1, RSV2, RSV3이 메모리 카드(2)에 제공되는 구조를 도시한다.

메모리 카드(2)의 개략적인 형태는 도 3을 참조하여 설명될 것이다.

메모리 카드(2)는, 합성 수지 등으로 제조되고 평면 형태의 장방형 및 작은 두께를 갖는 카드형 케이스(21); 케이스(21)내에 내장된 제어기(11) 및 플래시 메모리(12)로 구성된다. 메모리 카드(2)는 메모리 카드(2)를 장착하는 장착 메카니즘을 갖는 데이터 처리 장치(1)내에 장착되어 사용된다.

경사부 절단부(22)는 메모리 카드(2)의 케이스(21)의 선단부에 형성된다. 더욱이, 10개의 리세스(23)는 절단부(22)내에 형성된다. 리세스(23)는, 메모리 카드(2)가 데이터 처리 장치(1)의 장착 장치내에 장착될 때, 데이터 처리 장치의 접속 단자에 접속되게 배열된 외부 접속 단자를 갖는다. 즉, 메모리 카드(2)는 외부 접속 단자로서 기능하는 10개의 단자(24a, 24b, 24c, 24d, 24e, 24f, 24g, 24h, 24i,

24j)를 갖는다. 상기 외부 접속 단자는 3개의 데이터 라인 단자(24b, 24d, 24h), 전원용 4개의 전원 단자(24a, 24f, 24i, 24j), 및 3개의 리저브 단자(24c, 24e, 24g)로 구성된다.

에러 소거 방지 부재(25)는 메모리 카드(2)의 케이스(21)의 상부 표면에 인접된다. 에러 소거 방지 부재(25)는 케이스(21)내에 장착된 에러 소거 방지 스위치(20)에 사용된다. 에러 소거 방지 부재(25)가 마끄러질 때, 에러 소거 방지 스위치(20)는 스위칭 온/오프될 수 있다.

메모리 카드(2)에는 원호형의 제 1 고정용 절단부(26)가 제공되고 에러 소거 방지 스위치(20)의 어느 한 면상에 제공되며, 메모리 카드(2)가 데이터 처리 장치(1)의 장착 장치에 장착된후, 데이터 처리 장치(1)로부터 메모리 카드(2)의 분리를 방지한다. 더욱이, 장방형의 제 2 고정용 절단부(27)에는 에러 소거 방지 스위치(20)의 또다른 면내에 형성된다. 메모리 카드(2)가 데이터 처리 장치(1)의 장착 장치에 장착될 때, 고정용 절단부(26, 27)는 메모리 카드(2)의 분리를 방지하기 위해 데이터 처리 장치(1)의 장착 장치에 사용된다.

도 3에 도시된 메모리 카드(2)는 본 발명에 적용된 외부 기억 장치의 예이다. 즉, 본 발명은 외부 기억 장치의 형태에 달려 있지 않다. 본 발명은 외부 기억 장치의 형태에 상관없이 외부 기억 장치중의 임의의 한 장치에 적용될 수도 있다.

메모리 카드(2)에 장착된 플래시 메모리(12)의 기억 영역의 구조가 이제 설명될 것이다. 이제 설명될 기억 영역의 구조는 본 발명이 적용된 외부 기억 장치의 기억 영역의 구조의 예이다.

본 발명은 데이터 소거용 장치인 다수의 블록으로 분할된 기억 영역 및 기억된 부팅 데이터가 제공되는 부트 블록을 갖는 구조의 여러 외부 기억 장치에 광범위하게 기억 영역의 구조는 다음의 구조에 한정되는 것은 아니다.

도 4의 (a)에 있어서, 플래시 메모리(12)의 기억 영역은 데이터를 소거하기 위한 장치인 다수의 블록으로 분할된다. 메모리 카드(2)가 부팅될 때 데이터 처리 장치(1)에 의해 먼저 판독되도록 배열된 데이터인 부트 데이터를 기억하기 위한 부트 블록; 및 임의의 데이터가 기록된 데이터 블록으로 구성된다. 각각의 블록은 특정한 물리적 어드레스를 갖는다. 블록들은 파일을 관리하기 위한 최소 장치 및 데이터 소거용 장치이다. 즉, 파일은 한 개 이상의 블록내에 기억된다. 따라서, 한 개의 블록은 다수의 파일에 의해 사용될 수 없다.

각각의 부트 블록은 상태 1 및 상태 0인 두 상태중의 한 상태가 될 수 있는 다수의 비트로 구성된다. 따라서, 비트 단위의 변경은 1로부터 0로만 허용된다. 즉, 1 및 0로 구성된 데이터가 기록될 때, 1에 대응하는 비트는 유지되고 0에 대응하는 비트는 1로부터 0로 변경된다.

기록된 데이터가 소거될 때, 블록 단위로 일괄적으로 초기화 처리를 행하여, 상기 블록의 모든 비트를 1로 한다. 결과적으로, 블록상에 기록된 데이터는 일괄적으로 소거된다. 따라서, 데이터는 다시 상기 블록상에 기록될 수 있다.

본 발명은 각각의 비트가 단지 2개 상태가 될 수 있는 플래시 메모리(이른바, 2진형 플래시 메모리)뿐만 아니라, 3개 이상의 상태중 임의의 한 상태가 될 수 있는 플래시 메모리(이른바, 다중 값 플래시 메모리)에 적용될 수도 있다.

플래시 메모리(12)의 각각의 블록은, 도 4의 (b)에 도시된 바와같이, 데이터를 기록 또는 판독하는 단위인 다수의 페이지로 구성된다. 즉, 데이터가 플래시 메모리(12)상에 기록될 때, 페이지 버퍼(15)로부터 페이지 단위로 전송된 데이터는, 플래시 메모리 인터페이스 시퀀서(14)에 의해 플래시 메모리(12)상에 페이지 단위로 기록된다. 데이터가 플래시 메모리(12)로부터 판독될 때, 플래시 메모리 인터페이스 시퀀서(14)는 페이지 버퍼(15)에 데이터를 전송하도록 페이지 단위로 데이터를 판독한다.

각각의 페이지는 데이터 영역 및 용장 영역을 갖는다. 데이터 영역은 임의의 데이터가 기록된 영역이다. 용장 데이터는 데이터 영역상에 기록된 데이터를 관리하는데 필요한 정보가 기억된 영역이다.

구체적으로, 이른바 분산 관리 정보는 도 4의 (c)에 도시된 대로 블록을 관리하는데 필요한 정보로서 블록의 선두 페이지상의 용장 영역내에 기억된다. 블록의 제 2페이지로부터 페이지의 용장 영역내에 예비의 분산 관리 정보가 기억된 선두 페이지의 용장 영역에 기억된 것과 동일하다. 이른바, 추가의 분산 관리 정보는, 단지 분산 관리 정보에 의해 관리될 수 없는 추가의 정보로서 최종 페이지의 용장 영역내에, 분산 관리 정보를 대신해 기억된다.

상기 설명된 대로, 분산 관리 정보는 상기 플래시 메모리(12)내의 각각의 용장 영역내에 기억된다. 분산 관리 정보는 분산 관리 정보가 기억된 블록을 관리하는 정보이다. 분산 관리 정보에 따라, 예를 들어, 이전의 블록이 파일의 선두 블록인지를 나타내는 정보 및 파일이 다수의 블록으로 구성될 때 블록의 연결을 나타내는 정보가 얻어질 수 있다. 분산 관리 정보는 후술될 것이다.

메모리 카드(2)내의 블록의 분산 관리 정보는, 플래시 메모리의 전체 구조를 관리하는 정보로서 기능하는 집합 관리 정보를 발생하도록 집합지워진다. 그후, 집합 관리 정보는 파일로서 플래시 메모리(12)내에 기억된다.

보통, 집합 관리 정보는 각각의 블록에 대해 액세스하는데 요구된 정보를 얻는데 사용된다. 즉, 데이터가 데이터 처리 장치(1)와 메모리 카드(2)간에서 통신할 때, 데이터 처리 장치(1)는 내부 메모리(4)내에서 동일하게 발생하도록 메모리 카드(2)로부터 집합 관리 정보를 판독한다. 집합 관리 정보에 따라, 데이터 처리 장치(1)는 메모리 카드(2)에 대해 액세스한다. 결과적으로, 각각의 블록내에 기억된 분산 관리 정보에 대해 액세스하는 필요성이 생략될 수 있다. 이와같이, 보다 고속의 데이터 액세스가 허가된다.

분산 관리 정보, 추가의 분산 관리 정보, 및 집합 관리 정보가 후술될 것이다.

분산 관리 정보는 분산 관리 정보가 기억된 블록을 관리하는 정보이며, 분산 관리 정보는 16 바이트 용

장 영역상에 기록된다. 구체적으로, 분산 관리 정보는, 도 5에 도시된 바와같이, 1 바이트의 허가/금지 플래그, 1 바이트 블록 플래그, 4 비트 최종 플래그, 4 비트 참조 플래그, 1 바이트 관리 플래그, 2 바이트 논리 어드레스, 2 바이트 연결 플래그, 3 바이트 리저브 영역, 2 바이트 분산 관리 정보 여러 정정 부호 및 3 바이트 데이터 여러 정정 부호로 구성된다.

허가/금지 플래그는 블록이 사용 허가 상태 또는 사용 금지 상태인지를 나타내는 플래그이다. 구체적으로, 허가/금지 플래그는 사용 허가 및 사용 금지인 두 상태중의 한 상태를 나타낼 수 있다. 사용 허가는 블록이 사용될 수 있는 상태를 나타내고, 반면에, 사용 금지는 블록이 사용될 수 없는 상태를 나타낸다. 복구가 가능하지 않은 예러가 블록내에서 발생된다면, 허가/금지 플래그는 블록의 사용이 금지되도록 사용 금지되도록 설정된다.

블록 플래그는 블록의 상태를 나타내는 플래그이다. 구체적으로, 블록 플래그는 미사용, 선두 사용, 사용, 미소거의 4개의 상태중의 임의의 한 상태를 나타낸다. 미사용은 블록이 미사용 상태이거나 데이터가 블록으로부터 소거된 상태를 나타낸다. 이와같이, 미사용의 상태에서의 블록은 데이터가 즉시 기록될 수 있는 초기 상태(모든 비트는 1)이다. 선두 사용은 블록이 파일의 선두에서 사용되는 상태를 나타낸다. 그 내부에 기억된 부트 데이터를 갖는 부트 블록에서, 블록 플래그는 선두 사용상태이다. 사용은 선두를 제외한 부분에서 블록이 사용되는 상태를 나타낸다. 블록 플래그가 사용 상태일 때, 블록은 또다른 블록에 연결된다. 미소거는 블록상에 기록된 데이터가 불필요하게 되는 상태를 나타낸다. 예를 들어, 데이터가 소거될 때, 블록 플래그는 초기에 미소거 상태가 된다. 만일, 긴 처리 시간이 허용된다면, 미소거 상태에 있는 블록 플래그를 갖는 블록들이 소거된다. 결과적으로, 소거 처리는 효율적으로 실행된다.

최종 플래그는 파일이 종료되었는지를 나타내는 플래그이다. 구체적으로, 최종 플래그는 블록 연속 또는 블록 최종을 나타낸다. 블록 연속은 블록이 다음 블록에 연속인 상태를 나타낸다. 즉, 블록 연속은 블록에 기억된 파일이 다음의 부분을 갖고 이로써 파일이 또다른 블록에 이어지는 것을 나타낸다. 블록 종료는 블록이 최종 블록임을 나타낸다. 즉, 블록 종료는 블록내에 기억된 파일이 그 블록에서 종료됨을 나타낸다.

참조 플래그는 추가의 관리 정보의 참조를 지정하기 위한 플래그이다. 구체적으로, 참조 플래그는 참조 정보가 아님 및 참조 정보 있음의 두 상태중의 한 상태를 나타낸다. 참조 정보가 아님은 블록의 최종 페이지상의 용장 영역내에 유효 정보가 존재하지 않음을 나타낸다. 참조 정보 있음은 블록의 최종 페이지의 용장 영역에, 유효한 추가 관리 정보가 존재함을 나타낸다.

관리 플래그는 블록의 속성을 나타내는 플래그이다. 예를 들어, 관리 플래그는 블록이 판독 전용 블록 또는 기록을 허가하는 블록인지를 나타낸다. 관리 플래그는 또한 부트 블록인지 데이터 블록인지를 나타낸다.

논리 어드레스는 그 문자적 의미로 블록의 논리 어드레스를 나타낸다. 논리 어드레스의 값은 데이터가 재기록될 때 필수적으로 갱신된다. 논리 어드레스의 값은 논리 어드레스의 값이 다수의 블록에 의해 동시에 소유되지 않도록 하는 방식으로 설정된다.

연결 어드레스는 블록에 연결된 블록의 논리 어드레스이다. 즉, 블록에 기억된 파일은 다음의 부분을 갖는다. 만일, 파일이 또다른 블록에 연결된다면, 파일의 다음 부분이 기억된 다음의 블록의 논리 어드레스의 값은 연결 어드레스에 설정된다.

분산 관리 정보 여러 정정 부호는 관리 플래그, 논리 어드레스, 연결 어드레스, 및 분산 관리 정보 가운데 리저브 영역상에 기록된 데이터를 정정하는 여러 정정 부호이다. 각각의 허가/금지 플래그에서, 블록 플래그, 최종 플래그, 참조 플래그는 분산 관리 정보 여러 정정 부호에 의해 정정되지 않는다. 따라서, 허가/금지 플래그, 블록 플래그, 최종 플래그 및 참조 플래그는 분산 관리 정보 여러 정정 부호의 필요 없이 재기록될 수 있다.

데이터 여러 정정 부호는 데이터 여러 정정 부호가 기억된 페이지의 데이터 영역상에 기록된 데이터를 정정하는 여러 정정 부호이다.

분산 관리 정보 여러 정정 부호 및 데이터 여러 정정 부호는 메모리 카드(2)내에 포함된 여러 정정 회로(16)에 의해 사용된다. 따라서, 여러 정정 부호를 사용한 여러 정정은 데이터 처리 장치(1)에 따르지 않고 메모리 카드(2)에 따른 임의의 방법이 활용될 수도 있다.

추가 관리 정보는 블록의 최종 플래그상에 16 바이트 용장 영역내에 기억된 정보 및, 분산 관리 정보에 의해서만 관리될 수 없는 추가의 정보를 포함하는 정보이다.

구체적으로, 추가의 관리 정보는, 도 6에 도시된 대로, 1 바이트 허가/금지 플래그, 1 바이트 블록 플래그, 4 비트 최종 플래그, 4 비트 참조 플래그, 1 바이트 식별 번호, 2 바이트 유효 데이터 크기, 5 바이트 리저브 영역, 2 바이트 추가의 관리 정보 여러 정정 부호, 및 3 바이트 데이터 여러 정정 부호로 구성된다.

허가/금지 플래그, 블록 플래그, 최종 플래그, 참조 플래그, 리저브 영역, 및 데이터 여러 정정 부호는 분산 관리 정보의 경우와 유사하게 배열된다. 추가의 관리 정보 여러 정정 부호는 분산 관리 정보의 분산 관리 정보 여러 정정 부호에 대응한다. 추가의 관리 정보 여러 정정 부호는 식별 번호, 유효 크기, 및 추가의 관리 정보중의 리저브 영역을 정정하는 여러 정정 부호이다.

식별 번호 및 유효 데이터 크기는 분산된 관리 정보에 의해서만 관리될 수 없는 추가의 정보로서, 추가의 관리 정보에 포함된다.

식별 번호는 여러 해결 처리를 실행하는 정보이며, 식별 번호의 값은 블록내의 데이터가 재기록될 때마다 증가된다. 만일, 예러가 생기고, 이로써, 동일한 논리 어드레스를 갖는 다수의 블록이 존재한다면, 식별 번호는 블록상의 데이터가 새로운 것인지 오래된 것인지를 결정하는데 사용된다. 1 바이트 영역은 식별 번호에 의해 사용되는데, 식별 번호의 값은 0 내지 255이다. 식별 번호의 초기 값은 0이다. 만일, 동일한 논리 어드레스를 갖는 다수의 데이터 블록이 존재한다면, 식별 번호를 갖는 데이터 블록(그 값은

더욱 작음)은 유효하게 된다. 만일, 동일한 논리 어드레스를 갖는 다수의 부트 블록이 후술된 대로 존재한다면, 식별 번호를 갖는 부트 블록(그 번호가 더욱 큼)은 유효하게 된다.

유효 데이터 크기는 블록내의 유효 데이터의 크기이다. 즉, 만일, 블록의 데이터 영역이 플래그 부분을 갖는다면, 데이터 영역상에 기록된 데이터의 크기를 나타내는 값은 유효 데이터 크기로서 설정된다. 이때, 분산 관리 정보에서의 참조 플래그는 관리 정보가 있음으로 설정된다. 만일, 블록의 데이터 영역이 플래그 부분을 갖는다면, 0xffff는 데이터 영역이 플래그 부분을 갖지 않음을 나타내는 값으로서 유효 데이터 크기에 설정된다.

분산 관리 정보 및 상기의 추가의 관리 정보는 블록내의 데이터가 갱신될 때마다 최후 정보가 되도록 갱신된다.

집합 관리 정보는 각각의 블록의 분산 관리 정보에 의해 발생된 정보이며, 집합 관리 정보는 플래시 메모리(12)내에 파일로서 기억된다. 즉, 도 7에 도시된 대로, 모든 블록을 일괄적으로 관리하는 정보인 집합 관리 정보의 확인은 각각의 블록의 분산 관리 정보로부터 발생된다. 집합 관리 정보는 소정의 블록의 데이터 영역내에 기억된다. 집합 관리 정보가 한 개 또는 다수의 블록내에 기억될 수도 있다. 데이터 처리 장치(1)는 집합 관리 정보에 따라 각각의 블록에 대해 액세스하도록 요구된 정보를 얻는다.

메모리 카드(2)가 부트 업될 때, 부트 데이터는 데이터 처리 장치(1)에 의해 부트 블록으로부터 초기에 판독된다. 부트 데이터는 명령된 블록(집합 관리 정보가 기억됨)의 물리적 어드레스를 포함한다. 따라서, 데이터 처리 장치(1)는 내부 메모리(4)내의 집합 관리 정보를 발생하도록 물리적 어드레스에 대응하는 블록내에 기억된 집합 관리 정보를 판독한다. 집합 관리 정보에 따라, 데이터 처리 장치(1)는 메모리 카드(2)에 대해 액세스한다. 만일, 집합 관리 정보가 플래시 메모리(12)로부터 판독될 수 없다면, 데이터 처리 장치(1)는 모든 블록의 분산 관리 정보를 판독하여, 집합 관리 정보를 사용하기 위해 집합 관리 정보를 재구성한다.

데이터가 재기록될 때마다, 데이터 처리 장치(1)는, 요구의 상속에 따라, 집합 관리 정보가 플래시 메모리(12)에 일치하도록 내부 메모리(4)에서 발생된 집합 관리 정보를 갱신한다(즉, 집합적 관리 정보는 분산 관리 정보의 내용에 일치함). 한편, 플래시 메모리(12)내에 파일로서 기억된 집합 관리 정보는, 데이터 등의 기록이 실행될 때마다, 갱신되지 않는다. 집합 관리 정보의 내용은 적정 시간에서 일괄적으로 갱신된다. 예를 들어, 적정 시간은 전원이 인터럽트되기 전의 순간, 메모리 카드(2)에 대한 액세스가 소정의 시간보다 긴 시간 주기동안 이루어지는 순간, 또는 데이터의 기록이 소정 횟수보다 작지않게 실행되는 순간이다.

일반적으로, 플래시 메모리(12)에 대해 데이터를 재기록하도록 허가된 횟수는 상한 제한을 갖는다. 플래시 메모리(12)내에 파일로서 기록된 집합 관리 정보의 재기록이 다소 집합적으로 실행될 때, 집합 관리 정보가 기억된 블록의 재기록 횟수는 감소될 수 있다. 이와같이, 메모리 카드(2)의 수명은 연장될 수 있다.

본 발명의 필수 부분인 부트 블록의 처리가 후술될 것이다.

메모리 카드(2)가 부트 업될 때 부트 데이터는 데이터 처리 장치(1)에 의해 먼저 판독된 데이터이다. 부트 데이터는 메모리 카드(2)에 대한 액세스에 요구된 정보를 포함한다. 따라서, 부트 데이터가 기억된 부트 데이터내에, 에러가 있고 이로써 부트 데이터가 판독될 수 없다면, 메모리 카드(2)에 대한 액세스가 이루어질 수 없다. 이와같이, 부트 블록은 충분한 신뢰도를 가져야 한다.

따라서, 본 발명에 따른 메모리 카드(2)는 플래시 메모리(12)의 선두에서의 2개의 유효 블록내에 항상 기억되는 구조를 갖는다. 유효 블록은 사용 허가에 의해 상태이다. 즉, 유효 블록은 사용 허가에 설정된 허가/금지 플래그를 갖는 블록이다. 즉, 부트 데이터가 블록내에 기억될 때, 유효하지 않은 블록(즉, 블록은 각각 사용 금지에 설정된 허가/금지 플래그를 가짐)은 사용되지 않고 건너뛰어진다. 이와같이, 부트 데이터는 부트 블록이 발생되도록 유효 블록내에 기억된다.

상기 설명된 대로, 본 발명에 따른 메모리 카드(2)는 고정된 특정 블록이 부트 블록이 되도록 하는 구조를 갖는다. 만일, 유효하지 않은 블록이 된다면, 부트 데이터를 기억하는 블록이 변경된다. 이와같이, 부트 데이터는 2개의 유효 블록내에 기억된다. 즉, 에러가 선두 블록에서 이루어질지라도, 부트 데이터는 항상 2개의 유효 블록내에 이중으로 기억된다. 따라서, 본 발명에 따른 메모리 카드(2)의 신뢰도는 상당히 개선될 수 있다.

더욱이, 본 발명에 따른 메모리 카드(2)는 추가의 관리 정보내에 포함된 식별 번호가 데이터 블록의 경우와 부트 블록의 경우사이에서 다르게 사용되는 구조를 갖는다.

데이터 블록의 경우, 만일, 동일한 논리 어드레스를 갖는 다수의 블록이 존재한다면, 블록내에 기억된 새롭거나 오래된 데이터는 식별 번호를 사용하여 식별된다. 동일한 논리 어드레스를 갖는 다수의 데이터 블록이 주로 존재하는 이유는, 데이터 블록이 갱신될 때 에러가 존재하지 않는다는 것이다. 이때, 식별 번호를 갖는 데이터 블록내에 기억된 데이터(그 번호는 더욱 작음)는, 갱신전의 데이터이다. 따라서, 만일, 동일한 논리 어드레스를 갖는 다수의 데이터 블록이 존재한다면, 식별 번호를 갖는 데이터 블록(그 번호는 더욱 작음)은, 데이터가 갱신되기 전의 상태가 기억되도록 선택된다. 이와같이, 상기 데이터 블록내에 기억된 데이터는 유효 데이터로서 사용된다.

데이터 블록의 식별 번호는 논리 어드레스가 데이터 블록에 새롭게 지정될 때 0에 초기화된다. 식별 번호는 데이터 블록이 갱신될 때, 1 증가된다. 상술된 대로, 만일, 식별 번호가 255를 초과한다면, 식별 번호는 0가 된다.

부트 블록의 경우, 상기 설명된 대로, 식별 번호는, 플래시 메모리(12)의 선두에 2개의 유효 블록내에 기억된 새로운 또는 오래된 부트 데이터를 식별하는데 사용된다. 부트 데이터는 메모리 카드(2)에 대해 액세스하는데 요구된 정보이다. 따라서, 최후 정보가 사용되어야만 한다. 만일, 2개의 부트 블록의 식별 번호가 상호 다르다면, 부트 블록이 갱신될 때 에러가 이루어진다. 식별 번호를 갖는 부트 블록내에 기

억된 부트 데이터(그 값이 0)는, 새로운 정보를 포함하는 부트 데이터이다. 만일, 두 개의 부트 블록이 다른 식별 번호를 갖는다면, 식별 번호를 갖는 부트 블록(그 값이 더 큼)은, 상기 부트 블록내에 기억된 데이터가 부트 데이터로서 사용되도록 선택된다.

메모리 카드(2)는 포맷되어 처음 사용된다. 포맷팅 작업이 실행될 때, 부트 블록의 식별 번호는 0에 초기화된다. 식별 번호는 부트 블록내에 기억된 부트 데이터가 갱신될 때 1증가된다. 부트 블록의 경우, 식별 번호가 255를 초과한다면, 데이터 블록과 유사하게 0에 리턴된다.

플래시 메모리(12)의 선두에서 2개의 유효 블록이 부트 블록이 되고 식별 번호를 사용하여 새로운 또는 오래된 부트 블록을 관리하는 과정이 설명된다.

#### (1) 포맷 처리 과정

상술된 대로, 본 발명에 따른 메모리 카드(2)는 처음 사용될 때 포맷된다. 포맷 처리가 실행될 때, 부트 블록이 발생된다. 따라서, 포맷 처리가 실행될 때 부트 블록을 발생하는 과정이 설명될 것이다.

포맷 처리가 실행될 때, 데이터 처리 장치(1)는 메모리 카드(2)의 컨피규레이션 ROM(18)내에 기억된 정보를 우선 판독한다. 판독된 정보 등에 따라, 데이터 처리 장치(1)는 부트 블록내에 기억된 데이터(즉, 부트 데이터)를 발생한다. 부트 데이터로서 인식될 데이터를 판독하게 하기 위해, 판독 데이터가 부트 데이터임을 나타내는 정보(이후, 부트 식별자)는 부트 블록내에 기억되어야 하는 데이터내에 포함된다.

그후, 발생된 부트 데이터는, 부트 블록 및 여분의 부트 블록이 발생되도록, 플래시 메모리(12)의 2개의 선두 블록(최소한 물리적 어드레스 값을 갖는 블록 및 제 2 물리적 어드레스 값을 갖는 블록)상에 기록된다. 이때, 상기 부트 데이터가 기록된 블록의 추가의 관리 정보의 식별 번호는 0에 설정된다. 만일, 처리가 어떤 어려움 없이 완료된다면, 부트 블록을 발생하는 과정이 완료된다.

구체적으로, 부트 데이터를 기록하는 처리는 그 블록이 초기에 소거 처리되도록 실행된다. 그후, 부트 데이터는 이전의 블록상에 기록된다. 블록이 소거 처리될 수 없거나, 또는 부트 데이터가 소거 처리된 블록상에 기록된다면, 이전의 블록의 허가/금지 플래그는 사용 금지에 설정된다. 이와같이, 이전의 블록의 사용이 금지된다. 만일, 허가/금지 플래그가 사용 금지에 설정될 수 있다면, 허가/금지 플래그를 사용 금지로 설정하는 처리가 몇회 시도된다. 만일, 다수의 시도가 처리된 후에도 설정이 실행될 수 없다면, 데이터 처리 장치(1)는 매체가 비정상임을 판정한다. 이와같이, 데이터 처리 장치(1)는 메모리 카드(2)를 수용하지 않는다.

만일, 매체가 부트 블록이 되도록 의도된 블록내에 생긴다면, 본 발명에 따른 메모리 카드(2)는 사용될 수 있는 블록상의 부트 데이터를 유사하게 기록하도록 시도된다. 메모리 카드(2)는 동일한 내용을 갖는 두 개의 정상 부트 블록이 발생될 때까지 상기 처리를 계속한다. 결과로서, 동일한 내용을 갖는 부트 데이터는 플래시 메모리(12)의 2개의 선두 블록내에 각각 저장된다. 이와같이, 선두의 2개의 유효 블록은 부트 블록 및 여분의 부트 블록이 된다.

부트 데이터를 기록하는 처리는, 블록의 물리적 어드레스가 소정의 값 M에 도달한다면, 동일한 내용을 갖는 2개의 부트 블록이 발생되지 않는 경우에조차도 인터럽트된다. 소정의 값 M은 플래시 메모리(12)의 특성에 적합하게 이루어진다. 블록의 물리적 어드레스가 소정의 값 M에 도달할 때, 부트 블록을 발생하는 처리는, 한 개의 부트 블록이 발생할 때까지도 인터럽트된다. 블록의 물리적 어드레스가 소정의 값 M에 도달하고 부트 블록이 발생되지 않을 때까지도, 매체가 비정상임을 판정한다. 이와같이, 데이터 처리 장치(1)는 메모리 카드(2)를 수용하지 않는다.

일반적으로, 플래시 메모리(12)는 재기록 처리가 여러번 이루어진 블록에서 매체가 있을 가능성이 높다. 일단 매체가 있다면, 매체가 그 후에도 존재할 가능성이 있다. 따라서, 블록이 부트 블록으로서 사용되고 그 사용은 매체 때문에 금지되며, 블록은, 매체가 일시적인 어려움이라도 다시 사용되지 않는다. 상기 규칙이 적용되므로, 매우 중요한 정보인 부트 데이터는 더욱 충분한 신뢰도를 갖는 블록내에 기억된다. 이와같이, 부트 블록의 신뢰도가 개선될 수 있다. 상기 규칙이 선두 블록으로부터 제 2 부트 블록의 영역에만 적용되거나, 제 2 부트 블록이 존재하지 않는다면 상기 소정의 값 M에 도달한 물리적 어드레스를 갖는 블록까지의 영역에만 적용된다. 즉, 상기 언급된 규칙이 충분한 신뢰도를 갖는 부트 블록에 적용되고 동일한 적용은 다음의 블록(즉, 데이터 블록)에 금지된다. 결과로서, 기억 영역은 더욱 효율적으로 사용될 수 있다.

#### (2) 부트 블록을 갱신하는 과정

부트 데이터는 항상 동일 데이터인 것은 아니다. 메모리 카드(2)가 사용될 때, 부트 데이터의 내용은 변경되어야 한다. 따라서, 부트 블록은 때때로 부트 데이터를 재기록하도록 갱신된다. 부트 데이터를 재기록하는 처리는 다음의 규칙에 따라 실행된다.

부트 데이터가 재기록될 때, 집합 관리 정보의 내용은 변경되어야만 한다. 따라서, 플래시 메모리(12)내에 파일로서 기억된 집합 관리 정보는 부트 데이터가 재기록되기전에 유효하지 않은 것으로 된다. 집합 관리 정보가 다음에 사용될 때, 분산 관리 정보로부터의 재구성성이 실행된다.

부트 데이터의 내용이 재기록될 때, 부트 데이터가 기억되는 부트 블록은, 새로운 부트 데이터가 이전의 블록상에 기록됨으로써 소거 처리된다. 즉, 부트 데이터가 재기록될 때, 동일한 블록의 내용은 또다른 블록을 사용하지 않은채 갱신된다. 만일, 부트 블록을 갱신하는 처리가 실행되고 상기 블록의 사용이 금지될 때 매체가 있으면, 상기 처리는 실행되지 않는다.

새로운 부트 데이터에 대해 재기록하는 처리가 실행될 때, 작은 물리적 어드레스를 갖는 두 개의 발생된 블록들중의 한 블록이 처리된다. 이때, 갱신된 내용을 갖는 부트 블록의 식별 번호는 1증가된다. 만일, 재기록 작업전의 식별 번호의 값이 255가 되기전에, 그 값은 0가 된다.

여분의 부트 블록(즉, 큰 물리적 어드레스를 갖는 부트 데이터)이 갱신되고 이로써 또다른 부트 블록의 것과 동일한 내용을 갖도록 부트 데이터를 새로운 부트 데이터로 재기록하는 처리가 실행될 때, 여분의

부트 블록의 식별 번호는 또다른 부트 블록의 식별 번호의 것과 동일한 값을 갖게 된다.

상기 규칙에 따라 부트 블록을 갱신하는 처리는 도 8를 참조하여 후술될 것이다.

도 8에 도시된 실시예에서, 물리적 어드레스 1을 갖는 블록은 일반적으로 사용된 부트 블록(이후, 제 1 부트 블록이라 함)이며, 물리적 어드레스 2를 갖는 블록은 여분의 부트 블록(이후, 제 2 부트 블록이라 함)이다. 도 8에 도시된 실시예에서, 부트 데이터가 기록된 상태 및 판독될 수 있는 부트 데이터는 OK로 나타난다.

도 8의 (a)는 부트 블록을 재기록하는 처리가 실행되기전의 일반 상태를 도시한 것이다. 이때, 제 1 부트 블록의 식별 번호 및 제 2 부트 블록의 식별 번호는 동일 값을 갖는다. 구체적으로, 도 8의 (a)에서 도시된 실시예에서, 제 1 부트 블록의 식별 번호는 100이고, 제 2 부트 블록의 식별 번호는 100이다.

부트 데이터가 갱신될 때, 작은 물리적 어드레스를 갖는 부트 블록, 즉, 제 1 부트 블록은 도 8의 (b)에 도시된 대로, 부트 데이터를 재기록하도록 처리된다. 제 1 부트 블록이 부트 데이터를 재기록하는 처리이므로, 제 1 부트 블록은 제 1 부트 블록이 부트 블록으로서 사용될 수 없는 NG 상태가 된다.

메모리 카드(2)가 데이터 처리 장치(1)로부터 제거되거나 그 처리가 상기 언급된 상태에서 인터럽트된다면, 제 1 부트 블록은 재시작이 실행될 때 부트 블록으로서 사용된다. 상기 경우, 여분의 부트 블록인 제 2 부트 블록내에 기억된 부트 데이터는 메모리 카드(2)가 부트 업되도록 판독된다. 상기 경우, 제 1 부트 블록은 제 2 부트 블록으로부터 판독된 부트 데이터에 따라 재구성된다.

제 1 부트 블록이 갱신될 때, 제 1 부트 블록의 식별 번호는 도 8의 (c)에 도시된 바와같이 1 증가된다. 상기 실시예에서, 식별 번호의 값은 110이 된다. 상기 상태에서, 새로운 부트 데이터는 제 1 부트 블록에 기억되고 오래된 부트 데이터는 제 2 부트 블록내에 기억된다.

데이터 처리 장치(1)로부터의 메모리 카드(2)의 제거때문에, 처리가 상기 경우에 인터럽트될 때, 제 1 부트 블록내에 기억된 새로운 부트 데이터(즉, 큰 식별 번호를 갖는 부트 블록내에 기억된 부트 데이터)는 재시작이 실행될 때 판독된다. 이와같이, 메모리 카드(2)는 부트 업된다. 이때, 식별 번호를 갖는 부트 블록내에 기억된 부트 데이터(그 값은 더욱 작음(즉, 제 2 부트 블록))은, 식별 번호를 갖는 부트 블록(즉, 제 1 부트 블록)(그 값이 큼)내에 기억된 부트 데이터에 따라 갱신된다.

제 1 부트 블록을 갱신하는 처리가 완료될 때, 큰 물리적 어드레스를 갖는 부트 블록, 즉, 제 2 부트 블록은 도 8의 (d)에 도시된 것처럼 갱신된다. 이 때, 제 2 부트 블록은 부트 데이터가 재기록되기 때문에 부트 블록으로서 사용될 수 없는 NG 상태가 된다.

데이터 처리 장치(1)로부터 메모리 카드(2)의 제거가능함 때문에 처리가 인터럽트된다면, 제 1 부트 블록내의 부트 데이터는 재시작이 메모리 카드(2)를 부트 업을 실행하도록 사용된다. 그러나, 여분의 부트 블록이 존재하는 상태가 실현된다. 따라서, 여분의 부트 블록인 제 2 부트 블록은, 제 1 부트 블록으로부터 판독된 부트 데이터에 따라 재구성된다.

제 2 부트 블록이 갱신될 때, 제 2 부트 블록의 식별 번호는 도 8의 (e)에 도시된 대로, 1 증가된다. 상기 실시예에서, 식별 번호의 값은 110이 된다. 결과로서, 제 1 부트 블록의 식별 번호와 제 2 부트 블록의 식별 번호는 동일 값을 갖는다. 상기 언급된 처리의 결과로서, 제 1 및 제 2 부트 블록은 새로운 부트 블록이 기억되는 상태가 된다.

### (3)메모리 카드가 부트 업될 때 부트 블록을 판독하는 과정

부트 블록내에 기억된 부트 데이터는, 메모리 카드(2)가 데이터 처리 장치(1)에 연결되어 부트 업될 때, 데이터 처리 장치(1)에 의해 먼저 판독된다. 메모리 카드가 부트 업될 때 데이터 처리 장치(1)에 의해 부트 블록을 판독하는 과정이 설명될 것이다.

일반 상태에서, 동일 식별 번호 및 동일 내용을 갖는 2개의 부트 블록이 존재한다. 따라서, 상기 언급된 사실을 확인하는 처리는 메모리 카드가 부트 업될 때 항상 실행된다. 구체적으로, 부트 블록을 조사하기 위해 선두 블록에서 시작하는 순서로 다음의 처리가 실행된다.

- 선두 페이지의 분산 관리 정보가 정상적으로 판독될 수 있는지를 확인한다.
- 허가/금지 플러그가 사용 허가 상태인지를 확인한다.
- 블록 플러그가 선두 사용인지를 확인한다.
- 그 기억된 데이터가 부트 데이터인지를 확인하기 위해 부트 식별자가 검출된다.
- 최종 페이지상에 기억된 추가의 관리 정보가 정상적으로 판독될 수 있는지를 확인한다.
- 식별 번호가 판독된다. 제 1 부트 블록의 경우, 제 1 부트 블록의 식별 번호의 값이 기억된다. 제 2 부트 블록의 경우, 제 2 부트 블록의 식별 번호는 제 1 부트 블록의 것과 일치하는지를 확인한다.
- 부트 블록내의 데이터가 판독된다. 제 1 부트 블록의 경우, 판독된 데이터는 기억된다. 제 2 부트 블록의 경우, 제 1 부트 블록의 판독된 데이터 및 기억된 데이터가 상호 일치하는지가 확인된다.

상기 언급된 처리는 선두 블록에서 시작하도록 실행된다. 동일 식별 번호 및 동일 내용을 갖는 두 개의 부트 블록이 확인될 때, 부트 블록을 판독하는 동작이 완료된다. 상기 설명된 대로, 두 개의 부트 블록은 메모리 카드(2)의 신뢰도가 상당히 개선되도록 메모리 카드(2)가 부트 업될 때 확인된다.

물리적 어드레스가 소정의 값 M에 도달할 때까지 조사가 계속된후, 단지 한 개의 부트 블록이 존재한다면, 부트 블록내에 기억된 부트 데이터는 메모리 카드(2)를 부트 업하는데 사용된다. 만일, 선두 블록 내지 M번째 블록사이에 가용 블록이 존재한다면, 부트 데이터는 가용 블록상에 기록된다. 이와같이, 새로운 여분의 부트 블록이 발생된다. 만일, 선두 블록으로부터 M번째 블록까지 가용 블록이 존재하지 않는다면, 메모리 카드(2)는 단지 한 개의 부트 블록이 존재하는 상태에서 동작된다. 만일, 부트 블록이

N번째 블록까지 조사가 계속된후에도 존재하지 않는다면, 데이터 처리 장치(1)는 매체가 비정상임을 판정한다. 이와같이, 데이터 처리 장치(1)는 메모리 카드(2)를 수용하지 않는다.

만일, 부트 데이터가 정상적으로 두 블록상에 기록된 상태에서 조차도 식별 번호가 상호 다르다면, 큰 식별 번호를 갖는 블록은 유효 부트 블록으로서 선택된다. 메모리 카드(2)가 부트 업될 때, 큰 식별 번호를 갖는 블록내에 기억된 부트 데이터가 사용된다. 만일, 두 블록중의 한 블록의 식별 번호가 255이고, 그 다른 블록의 식별 번호가 0이라면, 식별 번호 0를 갖는 블록은 유효 부트 블록으로서 선택된다. 이와같이, 유효 부트 블록내의 부트 데이터가 사용된다. 부트 데이터가 정상적으로 단지 한 개의 블록상에 기록된다면, 상기 블록은 부트 블록으로서 사용된다.

메모리 카드(2)가 부트 업될 때 부트 블록을 판독하는 더욱 상세한 과정은, 도 9 내지 도 12에 도시된 흐름도를 참조하여 설명될 것이다. 상기 실시예에서, 변수 I, WPB, IDA, IDB가 사용된다. 변수 I, WPB는 물리적 어드레스가 입력되는 것인 반면에, 변수 IDA, IDB는 식별 번호의 값이 입력된 것이다.

메모리 카드가 부트업될 때, 0는 변수 단계 S1에서 변수 I에 입력된다. 더욱이, 0는 각각의 변수 IDA, IDB에 입력된다. 그후, 동작은 단계 S2에 진행된다.

단계 S2에서, 처리되어야 할 블록은 변수 I로 나타낸 물리적 어드레스를 갖는 블록이 된다. 그후, 동작은 단계 S3에 진행된다.

단계 S3에서, 처리되어야 할 블록으로부터, 분산 관리 정보를 판독 출력하는 지를 판정한다. 분산 관리 정보가 판독될 수 있으면 단계 S4로 진행하며, 분산 관리 정보가 판독될 수 없으면 단계 S25로 진행한다.

단계 S4에서, 처리되어야 할 블록의 허가/금지 플래그가 사용 허가 상태가 되었는지를 판정한다. 만일, 상태가 사용 허가 상태이면, 동작은 단계 S5로 진행된다. 만일, 상태가 사용 허가 상태가 아니면, 동작은 단계 S22로 진행된다.

단계 S5에서, 처리되어야 할 블록의 블록 플래그가 선두 사용인지가 판정된다. 만일, 상태가 선두 사용이면, 동작은 단계 S6으로 진행된다. 만일, 상태가 선두 사용이 아니면, 동작은 단계 S24로 진행된다.

단계 S6에서, 처리되어야 할 블록내에 기억된 데이터에 부트 식별자가 제공되었는지가 판정된다. 즉, 상기 데이터가 부트 데이터인지가 판정된다. 상기 데이터가 부트 데이터가 아니면, 동작은 단계 S7로 진행된다. 만일, 상기 데이터가 부트 데이터이면, 동작은 단계 S28로 진행된다.

단계 S7에서, 추가의 관리 정보가 처리 대상인 블록으로부터 판독될 수 있는지가 판정된다. 추가의 관리 정보가 판독될 수 있다면, 동작은 단계 S8로 진행된다. 추가의 관리 정보가 판독될 수 없다면, 동작은 단계 S25로 진행된다.

단계 S8에서, 처리 대상인 블록의 식별 번호의 값이 변수 IDA로 대체된다. 그후, 동작은 단계 S9로 진행된다.

단계 S9에서, 처리 대상인 블록으로부터 부트 데이터가 판독될 수 있는지를 판정한다. 만일, 부트 데이터가 판독될 수 있다면, 동작은 단계 S10으로 진행된다. 만일, 부트 데이터가 판독될 수 없다면, 동작은 단계 S25로 진행된다.

단계 S10에서, 부트 데이터는 판독된 부트 데이터를 기억하기 위해 처리되어야 할 블록으로부터 판독된다. 그후, 동작은 단계 S11로 진행된다.

단계 S11에서, 변수 I의 값은 변수 WPB에 대체되고 변수 I의 값은 1 증가된다. 그후, 동작은 단계 S12에 진행된다.

단계 S12에서, 처리 대상인 블록은 변수 I로 나타낸 물리적 어드레스를 갖는 블록이 되어야 한다. 그후, 동작은 단계 S13으로 진행된다.

단계 S13에서, 처리 대상인 블록으로부터 분산 관리 정보가 판독될 수 있는지가 판정된다. 분산 관리 정보가 판독될 수 있다면, 동작은 단계 S14로 진행된다. 만일, 분산 관리 정보가 판독될 수 없다면, 동작은 단계 S32에 진행된다.

단계 S14에서, 처리 대상인 블록의 허가/금지 플래그가 사용 허가 상태에 있는지가 판정된다. 만일, 상태가 사용 허가 상태이면, 동작은 단계 S15로 진행된다. 만일, 상태가 사용 허가 상태가 아니면, 동작은 단계 S29로 진행된다.

단계 S15에서, 처리되어야 할 블록의 블록 플래그가 선두 사용인지가 판정된다. 만일, 상태가 선두 사용 상태이면, 동작은 단계 S16으로 진행된다. 만일, 상태가 선두 사용이 아니면, 동작은 단계 S31로 진행된다.

단계 S16에서, 처리되어야 할 블록내에 기억된 데이터에 부트 식별자가 제공되었는지가 판정된다. 즉, 상기 데이터가 부트 데이터인지가 판정된다. 상기 데이터가 부트 데이터가 아니면, 동작은 단계 S17로 진행된다. 만일, 상기 데이터가 부트 데이터이면, 동작은 단계 S35로 진행된다.

단계 S17에서, 추가의 관리 정보가 처리 대상인 블록으로부터 판독될 수 있는지가 판정된다. 추가의 관리 정보가 판독될 수 있다면, 동작은 단계 S18로 진행된다. 추가의 관리 정보가 판독될 수 없다면, 동작은 단계 S32로 진행된다.

단계 S18에서, 처리 대상인 블록의 식별 번호의 값이 변수 IDB로 대체된다. 그후, 동작은 단계 S19로 진행된다.

단계 S19에서, 변수 IDA, IDB의 값이 상호 비교된다. 만일, 변수 IDA, IDB의 값이 동일하다면, 동작은 단계 S20으로 진행된다. 만일, 변수 IDA, IDB의 값이 동일하지 않다면, 동작은 단계 S32로 진행된다.

단계 S20에서, 처리 대상인 블록으로부터 부트 데이터가 판독될 수 있는지가 판정된다. 만일, 부트 데이터가 판독될 수 있다면, 동작은 단계 S21로 진행한다. 만일 부트 데이터가 판독될 수 없다면, 동작은 단계 S32로 진행한다.

단계 S21에서, 부트 데이터는 처리 대상인 블록으로부터 판독된다. 그후, 판독된 부트 데이터와 판독 및 기억된 부트 데이터가 상호 일치하는지가 판정된다. 부트 데이터 항목은 두 부트 블록으로부터 부트 데이터의 판독이 정상적으로 실행될 때, 상호 일치한다. 따라서, 처리가 완료된다. 만일, 부트 데이터 항목이 상호 일치하지 않는다면, 동작은 단계 S32에 진행한다.

허가/금지 플래그가 S4에서 사용 허가 상태가 아니라면, 동작은 상술된 대로 단계 S22로 진행한다.

단계 S22에서, 변수 I 및 변수 M의 값은 상호 비교된다. 만일, 변수 I의 값이 소정의 값 M보다 작다면, 동작은 단계 S23으로 진행한다. 변수 I의 값이 단계 S22에서의 소정의 값 M보다 작지 않다면, M번째의 블록까지 조사가 실행된후 부트 데이터가 얻어질 수 없는 경우가 있다. 이 경우, 예러가 있는 것으로 판정되어 처리를 종료한다.

단계 S23에서, 변수 I의 값은 1 증가된다. 그후, 동작은 단계 S2로 복귀하며 처리가 반복된다.

블록 플래그가 단계 S5에서 선두 사용이 되지 않는다면, 동작은 단계 S24로 진행한다. 단계 S24에서, 상기 블록 플래그가 미사용 상태인지가 판정된다. 만일, 상태가 미사용 상태이면, 동작은 도 11에 도시된 대로 단계 S36에 진행한다. 만일, 상태가 미사용이 아니면, 동작은 단계 S25로 진행한다.

단계 S25에서, 처리 대상이 되는 블록은 소거 처리된다. 그후, 동작은 단계 S26로 진행한다.

단계 S26에서, 단계 S25에서의 소거 처리가 정상적으로 완료되었는지가 판정된다. 소거 처리가 정상적으로 완료된다면, 동작은 도 11에 도시된 단계 S36로 진행한다. 소거 처리가 정상적으로 완료된다면, 동작은 단계 S27로 진행한다.

단계 S27에서, 처리 대상이 되는 블록의 허가/금지 플래그는 사용 금지 상태로 설정된다. 그후, 동작은 단계 S22로 진행하여 상기 처리가 실행된다.

데이터가 단계 S6에서의 부트 데이터가 아니라면, 동작은 상술된 대로 단계 S28로 진행한다.

단계 S28에서, 처리 대상이 되는 블록은 또다른 블록으로 시프트된다. 그후, 동작은 상기 처리가 실행되도록 단계 S25로 진행한다.

분산 관리 정보가 단계 S3에서 판독될 수 없다면, 추가의 관리 정보가 단계 S7에서 판독될 수 없거나 부트 데이터가 단계 S9에서 판독될 수 없다면, 동작은 단계 S25로 진행하여 상기 처리가 실행된다.

허가/금지 플래그가 단계 S14에서의 사용 허가 상태로 되지 않는다면, 동작은 단계 S29로 진행한다. 단계 S29에서, 변수 I의 값 및 소정의 값 M은 상호 비교된다. 만일, 변수 I의 값이 소정의 값 M보다 작다면, 동작은 단계 S30으로 진행한다. 제 2 부트가 M번째 블록에 대한 조사가 실행된후 제 2 부트 데이터가 얻어질 수 없을 때, 변수 I의 값은 단계 S29에서의 소정의 값 M보다 작지 않다. 이 경우, 부트 데이터를 판독하는 처리는 인터럽트된다. 판독 및 기억된 부트 데이터는 메모리 카드(2)가 부트 업되는데 사용된다.

단계 S30에서, 변수 I의 값은 1 증가된다. 그후, 동작은 단계 S12로 복귀되어 처리가 반복된다.

만일, 블록 플래그가 단계 S15에서 선두 사용 상태가 아니라면, 동작은 상기 설명된 대로 단계 S31로 진행한다. 단계 S31에서, 상기 블록 플래그가 미사용 상태인지가 판정된다. 만일, 상태가 미사용 상태이면, 동작은 도 12에 도시된 단계 S47로 진행한다. 만일, 상태가 미사용 상태가 아니면, 동작은 단계 S32로 진행한다.

단계 S32에서, 처리 대상이 되는 블록이 소거 처리된다. 그후, 동작은 단계 S33으로 진행한다.

단계 S33에서, 소거 처리가 단계 S32에서 정상적으로 완료되었는지가 판정된다. 만일, 소거 처리가 정상적으로 완료된다면, 동작은 도 12에서 도시된 단계 S47로 진행한다. 만일, 소거 처리가 정상적으로 완료되지 않는다면, 동작은 단계 S34로 진행한다.

단계 S34에서, 처리 대상이 되는 블록의 허가/금지 플래그는 사용 금지 상태로 설정된다. 그후, 동작은 단계 S29로 진행하여 상기 처리가 실행된다.

만일, 데이터가 단계 S16에서의 부트 데이터가 아니라면, 동작은 단계 S35로 진행한다. 단계 S35에서, 처리 대상이 되는 블록은 또다른 블록으로 시프트된다. 그후, 동작은 단계 S32로 진행하여 상기 처리가 실행된다.

만일, 분산 관리 정보가 단계 S13에서 판독될 수 없다면, 추가의 관리 정보가 단계 S17에서 판독될 수 없다면, 만일, 변수 10A의 값 및 변수 10B의 값이 단계 S19에서 상호 일치하지 않는다면, 부트 데이터가 단계 S20에서 판독될 수 없다면 또는 부트 데이터 항목이 단계 S21에서 상호 일치하지 않는다면, 동작은 단계 S32로 진행하여 상기 처리가 실행된다.

도 11에서, 변수 I의 값은 단계 S36에서 변수 W로 대체된다. 더욱이, 변수 I의 값은 증가된다. 그후, 동작은 단계 S37로 진행한다.

단계 S38에서, 분산 관리 정보가 처리 대상인 블록으로부터 판독될 수 있는지가 판정된다. 만일, 분산 관리 정보가 판독될 수 있다면, 동작은 단계 S39로 진행한다. 만일, 분산 관리 정보가 판독될 수 없다면, 동작은 단계 S45로 진행한다.

단계 S39에서, 처리 대상이 되는 블록의 허가/금지 플래그가 사용 허가인지가 판정된다. 만일, 상태가 사용 허가 상태이면, 동작은 단계 S40로 진행한다. 만일, 상태가 사용 허가 상태가 아니라면, 동작은 단

계 S45로 진행한다.

단계 S40에서, 처리 대상이 되는 부트 블록에 기억된 데이터에 블록 식별자가 제공되었는지가 판정된다. 즉, 상기 데이터가 부트 데이터인지가 판정된다. 만일, 상기 데이터가 부트 데이터이면, 동작은 단계 S41로 진행한다.

단계 S41에서, 추가의 관리 정보가 처리 대상이 되는 블록으로부터 판독될 수 있는지가 판정된다. 만일, 추가의 관리 정보가 판독될 수 있다면, 동작은 단계 S42로 진행한다.

단계 S42에서, 처리 대상이 되는 블록으로부터 부트 데이터가 판독될 수 있는지가 판정된다. 만일, 부트 데이터가 판독될 수 있다면, 동작은 단계 S43으로 진행한다.

단계 S43에서, 처리 대상이 되는 블록으로부터 부트 데이터가 판독되어 판독된 부트 데이터가 기억된다. 그후, 동작은 단계 S44로 진행한다.

단계 S44에서, 단계 S43에서 기억 및 판독된 부트 데이터는 변수 W8로 나타낸 물리적 어드레스를 갖는 블록상에 기록된다. 이때, 부트 데이터를 판독하는 처리가 인터럽트된다. 그후, 단계 S43에서 기억 및 판독된 부트 데이터는 메모리 카드(2)를 부트 업하는데 사용된다.

만일, 분산 관리 정보가 단계 S38에서 판독될 수 없거나 또는 허가/금지 플래그가 단계 S39에서 사용 허가상태에 있으면, 동작은 상술된 대로 단계 S45로 진행한다. 단계 S45에서, 변수 I의 값 및 소정의 값 M은 상호 비교된다. 만일, 변수 I의 값이 소정의 값 M보다 작으면, 동작은 단계 S46으로 진행한다. 변수 I의 값은 부트 데이터가 M번째 블록까지 조사가 실행된후 얻어질 수 없을 때 소정의 값 M보다 작지 않게 이루어진다. 이 경우, 에러로서의 판정이 실행되고 처리는 종료된다.

단계 S46에서, 변수 I의 값은 1 증가된다. 그후, 동작은 단계 S37로 복귀하고 처리는 반복된다.

만일, 데이터가 단계 S40에서의 부트 데이터가 아니라면, 만일, 추가의 관리 정보가 단계 S41에서 판독될 수 없다면, 또는 부트 데이터가 단계 S42에서 판독될 수 없다면, 에러로서 판정이 실행되고 처리가 종료된다. 상기 경우는 부트 데이터가 두 부트 블록중의 한 블록으로부터 판독될 수 없을 때 발생한다.

도 12에 도시된대로, 변수 I08의 값이 0xffff인지가 단계 S47에서 판정된다. 만일, 변수 I08의 값이 0xffff이 아니라면, 동작은 단계 S48로 진행한다. 변수 I08의 값이 0xffff이면, 동작은 단계 S49로 진행한다.

단계 S48에서, 변수 I0A의 값과 변수 I08의 값에 1증가함으로써 얻어진 값은 상호 비교된다. 만일, 두 값이 동일하다면, 동작은 단계 S49로 진행한다. 만일, 두 값이 동일하지 않다면, 동작은 단계 S51로 진행한다.

단계 S49에서, 변수 I의 값은 변수 W8에 대체된다. 그후, 동작은 단계 S50으로 진행한다.

단계 S50에서, 판독 및 기억된 부트 데이터는 변수 W8로 나타낸 물리적 어드레스를 갖는 블록상에 기록된다. 이때, 부트 데이터에 대한 처리는 완료되고 판독 및 기억된 부트 데이터는 메모리 카드(2)를 부트 업하는데 사용된다.

단계 S51에서, 변수 I08의 값과 변수 I0A의 값에 1가산함으로써 얻어진 값은 상호 비교된다. 만일, 두 변수가 동일하다면, 동작은 단계 S50으로 진행하여 상기 처리가 실행된다.

변수 I08의 값과 변수 I0A의 값에 1가산함으로써 얻어진 값은, 두 부트 블록의 식별 번호가 동일하지 않을 때와 식별 번호가 순차 번호가 아닐 때, 단계 S51에서 동일하지 않다. 부트 데이터의 판독이 실행될 수 있으므로, 수동 복구 모드는 데이터 처리 장치(1)가 적절한 처리를 실행하도록 사용된다.

이와같이, 부트 블록을 판독하는 처리는 메모리 카드가 부트 업될 때 실행된다. 상기 언급된 처리가 실행됨으로써, 두 부트 블록은 메모리 카드(2)가 부트 업될 때 인식된다. 따라서, 메모리 카드(2)의 신뢰도는 상당히 개선될 수 있다.

#### 발명의 효과

상기 설명된 대로, 본 발명에 따라, 부트 데이터는 각각의 다수의 다른 블록에 기억된다. 따라서, 부트 데이터가 기억된 블록이 사용될 수 없을지라도, 또다른 블록에 기억된 부트 데이터는 외부 기억 장치를 부트 업하는데 사용될 수 있다.

새로운 또는 오래된 부트 데이터를 나타내는 식별 번호가 블록에 기억되므로, 최신 부트 데이터는 항상 사용될 수 있다. 즉, 본 발명에 따라, 혼합된 방식으로 존재하는 새로운 부트 데이터 및 오랜 부트 데이터에서의 상태가 실현될지라도, 최신 부트 데이터는 항상 사용될 수 있다. 따라서, 데이터의 일관성이 유지될 수 있다.

따라서, 본 발명에 따라, 에러의 발생은 부트 데이터가 판독될 때 만족스럽게 방지될 수 있다. 이와같이, 신뢰할 만한 외부 기억 장치가 제공될 수 있다.

소정의 정밀도로 본 발명의 양호한 형태로 개시될 지라도, 양호한 형태의 개시는 본 발명의 정신 및 범위를 벗어나지 않은채 상세한 구성 및 조합으로 변경될 수 있다.

#### (57) 청구의 범위

청구항 1. 데이터의 소거를 소정의 블록 단위로 행함과 함께, 부트 업시에 최초로 판독된 부트 데이터를 기억하는 블록을 갖는 외부 기억 장치에 있어서,

다수의 다른 블록에 부트 데이터를 각각 기억하고,

부트 데이터가 기억된 각 블록에 기억된 부트 데이터가 새로운 것인지 오래된 것인지를 나타내는 식별 번호가 각각의 블록에 기억되는 외부 기억 장치.

청구항 2. 제 1항에 있어서, 다수의 다른 블록내에 기억된 부트 데이터중에서 최신의 부트 데이터는, 상기 외부 기억 장치가 최신 부트 데이터를 사용하여 부트 업되도록 상기 외부 기억 장치가 부트 업될 때 식별 번호에 따라 판독되는 외부 기억 장치.

청구항 3. 제 1항에 있어서, 상기 외부 기억 장치가 부트 업될 때 식별 번호에 따라, 다수의 다른 블록내에 기억된 각각의 부트 데이터가 새로운 것인지 오래된 것인지를 판정하며, 부트 데이터는 오래된 부트 데이터가 존재하는 경우 최신 부트 데이터로 재기록되는 외부 기억 장치.

청구항 4. 소정의 블록 단위로 데이터를 소거하도록 배열된 외부 기억 장치가 부트 업될 때 최초 판독된 부트 데이터는 상기 외부 기억 장치내에 기억되는 데이터 처리 방법에 있어서,

다수의 다른 블록의 각각의 블록에 부트 데이터를 기억하는 단계; 및

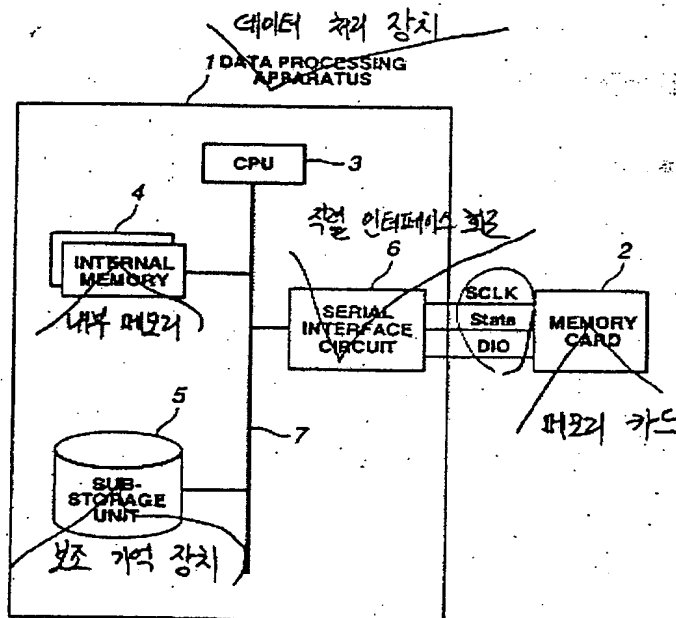
부트 데이터가 기억된 각각의 블록내에, 그 블록내에 기억된 부트 데이터가 새로운 것인지 오래된 것인지를 나타내는 식별 번호를 기억하는 단계를 구비하는 데이터 처리 방법.

청구항 5. 제 4항에 있어서, 상기 외부 기억 장치가 부트 업될 때, 다수의 다른 블록내에 기억된 부트 데이터들중에서 최신 부트 데이터가 식별 번호에 따라 판독되는 데이터 처리 방법.

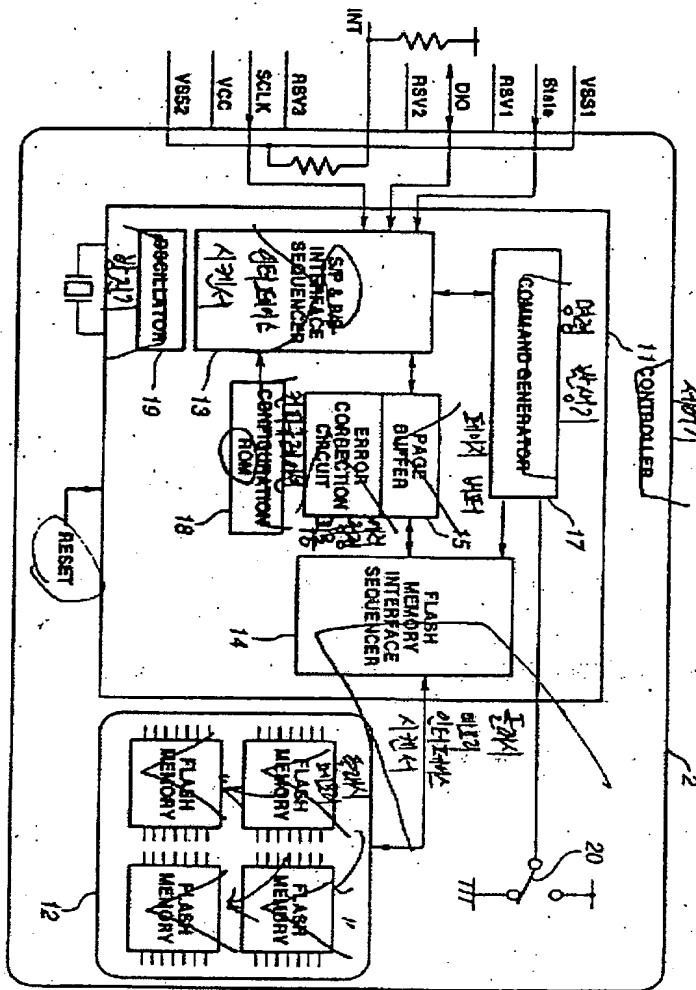
청구항 6. 제 4항에 있어서, 상기 외부 기억 장치가 부트 업될 때, 다수의 다른 블록내에 기억된 부트 데이터가 새로운 것인지 오래된 것인지를 식별 번호에 따라 판정하며, 오래된 부트 데이터가 존재할 때 오래된 부트 데이터는 최신 부트 데이터로 재기록되는 데이터 처리 방법.

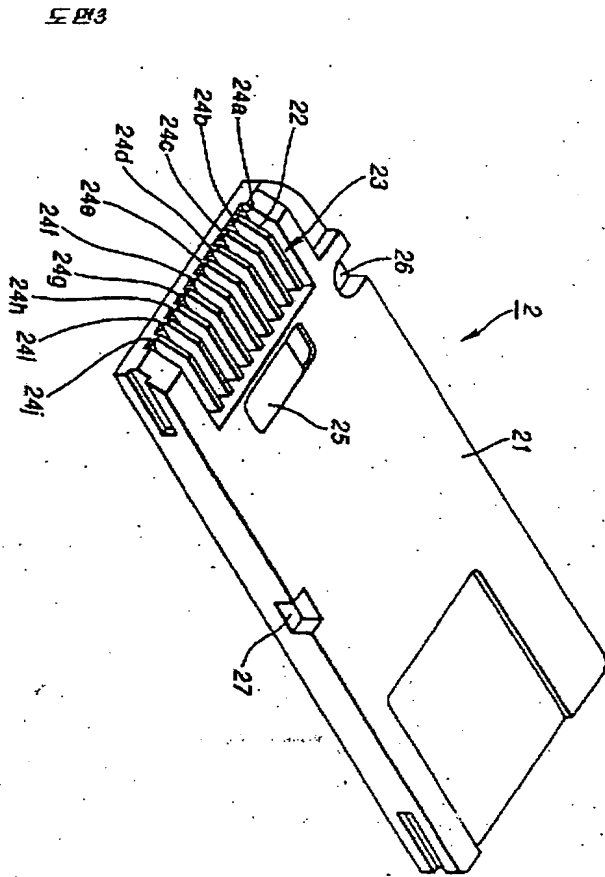
도면

도면1



도 2





도면 3

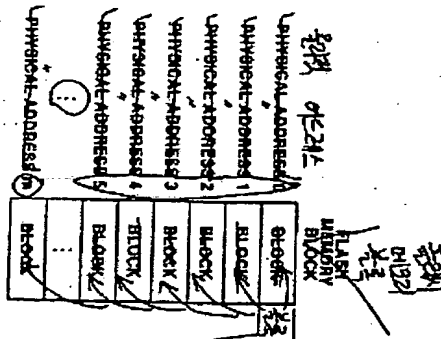


FIG. 4A  
(a)

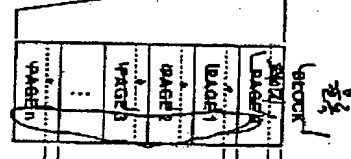


FIG. 4B  
(b)

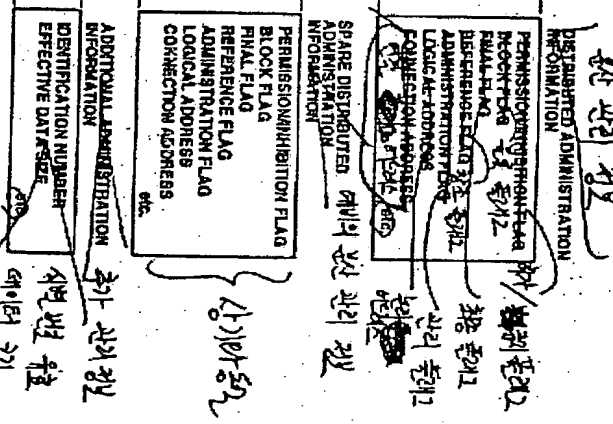
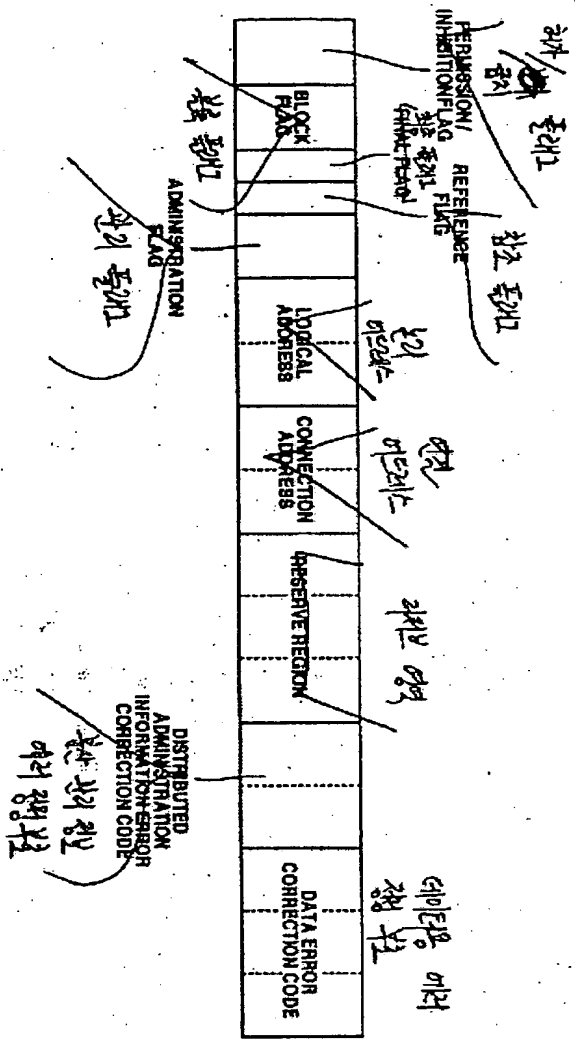
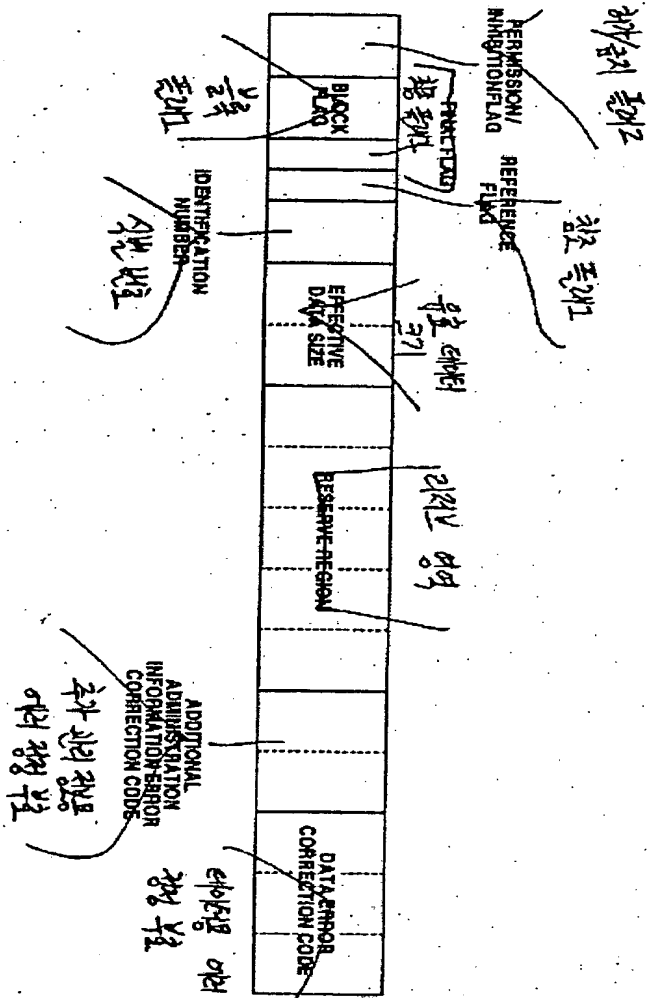


FIG. 4C  
(c)

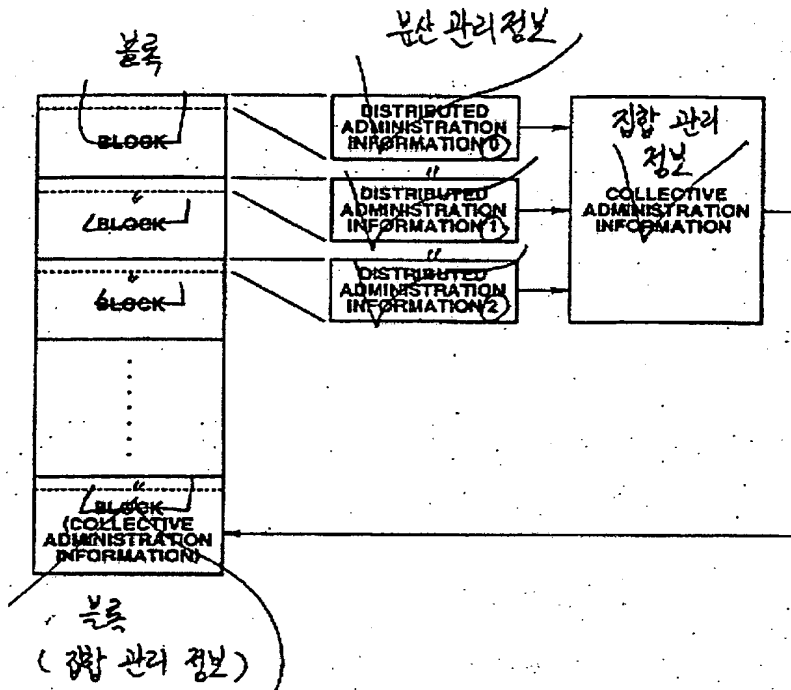
도면5



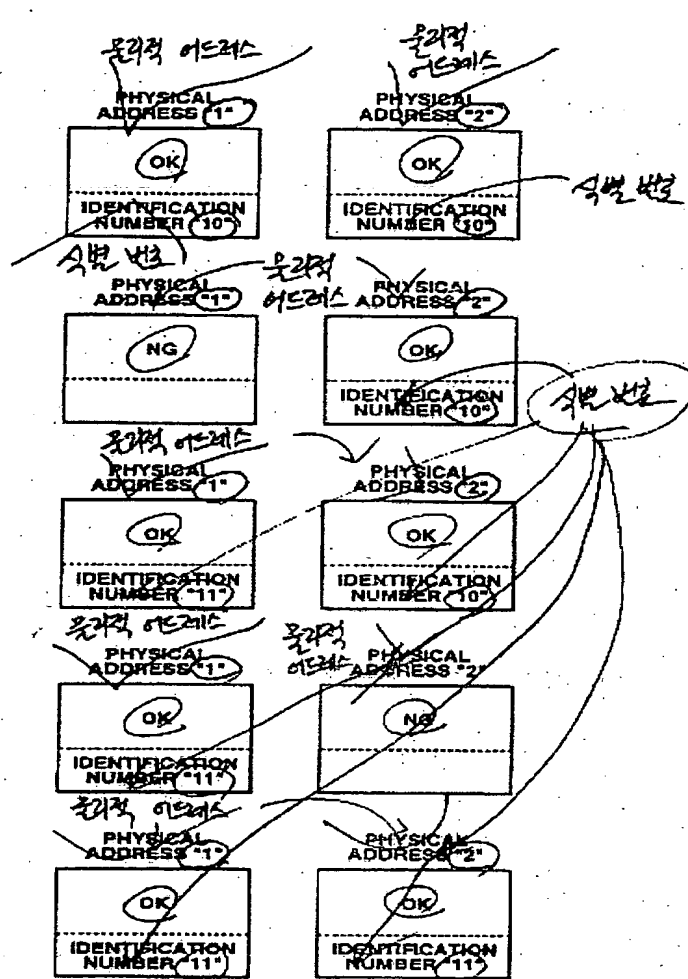
도 28



도면 7

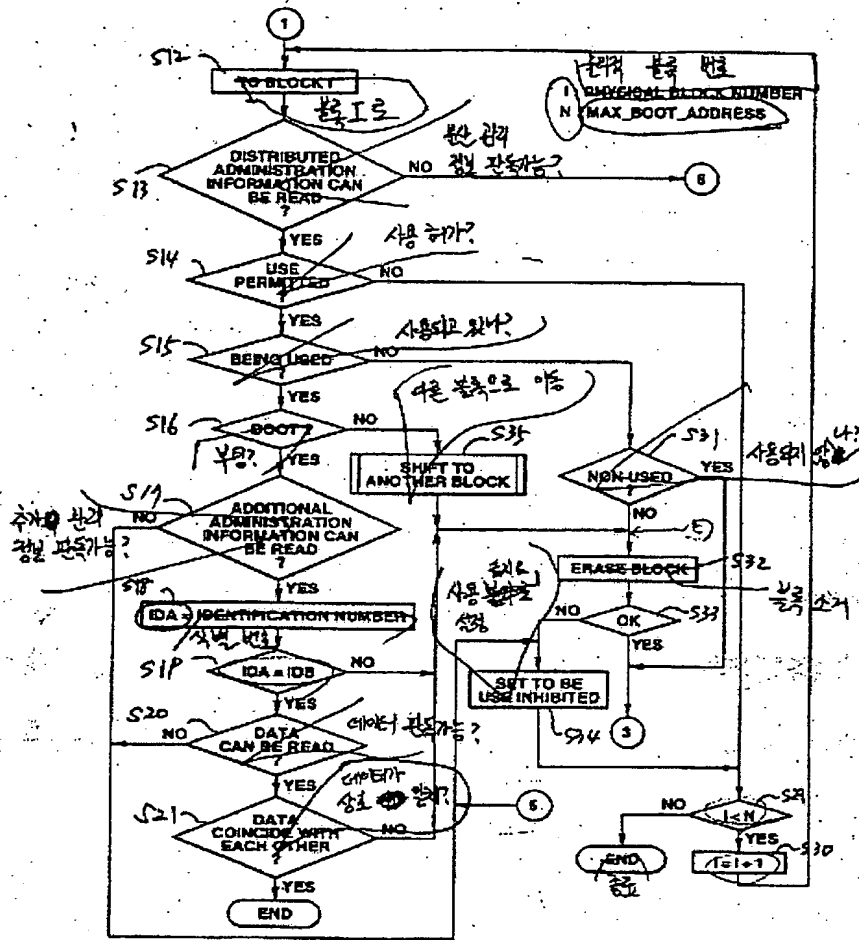


도면 8

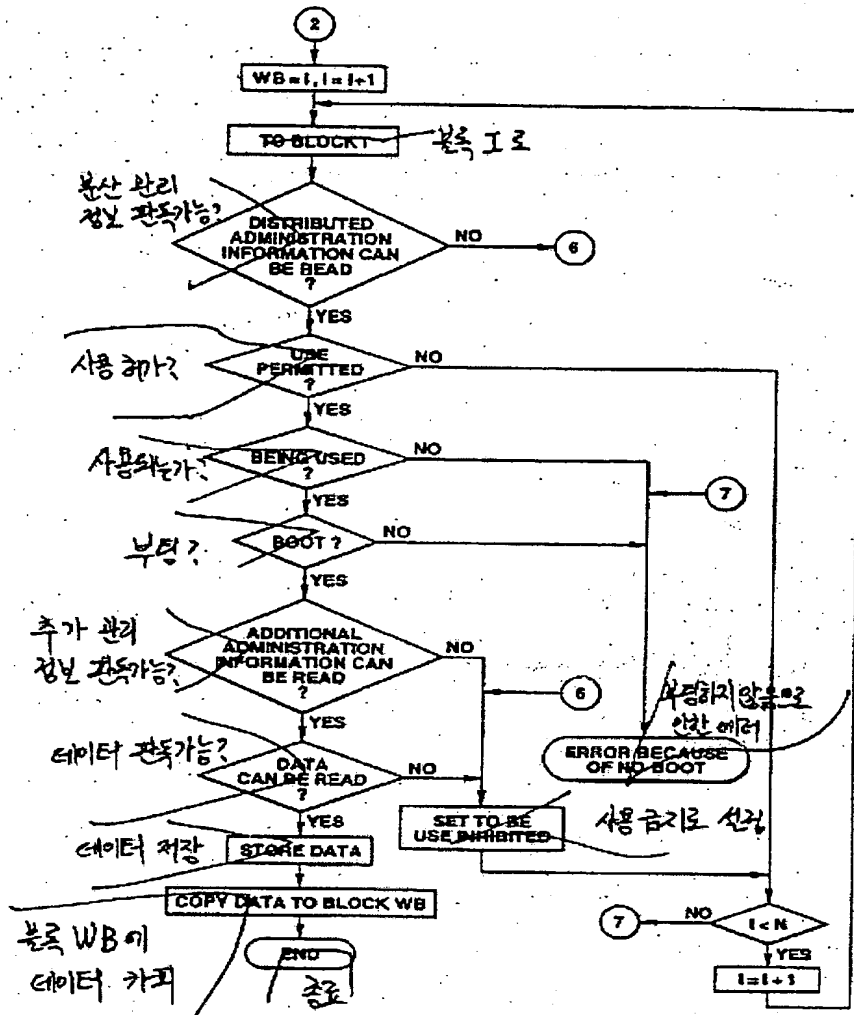




도면 10



도면 11



도면12

